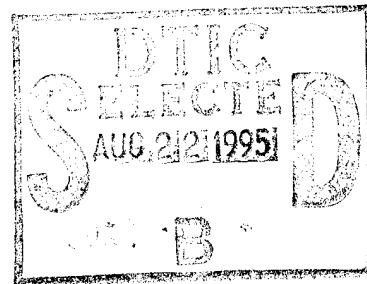


# NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA



## THESIS

**IMPLEMENTATION OF ERROR DETECTION  
AND CORRECTION (EDAC) IN THE STATIC  
RANDOM ACCESS MEMORY (SRAM) ABOARD  
PETITE AMATEUR NAVY SATELLITE (PANSAT)**

by

Craig R. Oechsel

March, 1995

Thesis Advisor:  
Second Reader:

Randy L. Wight  
Herschel H. Loomis, Jr.

**Approved for public release; distribution is unlimited.**

19950821 047

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE March 1995	3. REPORT TYPE AND DATES COVERED Master's Thesis		
4. TITLE AND SUBTITLE Implementation of Error Detection and Correction (EDAC) in the Static Random Access Memory (SRAM) Aboard Petite Amateur Navy Satellite (PANSAT)		5. FUNDING NUMBERS		
6. AUTHOR(S) Craig R. Oechsel				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000		8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING/MONITORING AGENCY REPORT NUMBER		
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words)  This thesis documents the design of a bus controller that provides EDAC capability to the SRAM of an Intel M80C186XL Microprocessor running at 7.3728 MHz. The system was designed for use during a two-year mission in a low earth orbit on board PANSAT. The system uses standard CMOS components together with the Harris ACS630MS EDAC circuit to provide dual-bit error detection with single-bit error correction. The single-bit error correction process is transparent to the microprocessor. All single-bit errors detected are automatically corrected in memory during the same bus cycle in which they are detected. The EDAC circuit computes the check bits based upon a 16-bit data word. Byte write capability is provided by using a "read-modify-write" method. The system was built on a wire wrap development board and tested for proper operation.				
14. SUBJECT TERMS PANSAT, Memory, Static Random Access Memory, SRAM, Error Detection and Correction, EDAC			15. NUMBER OF PAGES 96	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)  
Prescribed by ANSI Std. Z39-18 298-102

DEIC QUALITY INSPECTED 2



Approved for public release; distribution is unlimited.

IMPLEMENTATION OF ERROR DETECTION AND CORRECTION (EDAC)  
IN THE STATIC RANDOM ACCESS MEMORY (SRAM)  
ABOARD PETITE AMATEUR NAVY SATELLITE (PANSAT)

Craig R. Oechsel  
Lieutenant, United States Navy  
B.S., United States Naval Academy, 1987

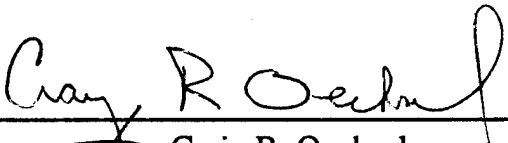
Submitted in partial fulfillment  
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

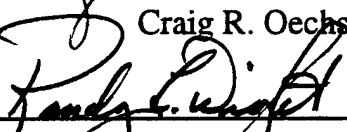
from the


NAVAL POSTGRADUATE SCHOOL  
March 1995

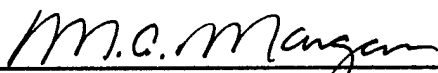
Author:

  
Craig R. Oechsel

Approved by:

  
Randy L. Wight, Thesis Advisor

  
Herschel H. Loomis, Jr., Second Reader

  
Michael A. Morgan, Chairman  
Department of Electrical and Computer Engineering

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



## **ABSTRACT**

This thesis documents the design of a bus controller that provides EDAC capability to the SRAM of an Intel M80C186XL Microprocessor running at 7.3728 MHz. The system was designed for use during a two-year mission in a low earth orbit on board PANSAT. The system uses standard CMOS components together with the Harris ACS630MS EDAC circuit to provide dual-bit error detection with single-bit error correction. The single-bit error correction process is transparent to the microprocessor. All single-bit errors detected are automatically corrected in memory during the same bus cycle in which they are detected. The EDAC circuit computes the check bits based upon a 16-bit data word. Byte write capability is provided by using a "read-modify-write" method. The system was built on a wire wrap development board and tested for proper operation.



# TABLE OF CONTENTS

I. INTRODUCTION .....	1
A. PURPOSE .....	1
B. SCOPE OF THESIS .....	1
II. BACKGROUND INFORMATION .....	3
A. MICROSATELLITE DESIGN PHILOSOPHY .....	3
B. PANSAT MISSION LIFE AND OPERATING ENVIRONMENT .....	3
1. Thermal Environment .....	4
2. Radiation Environment .....	4
C. RADIATION EFFECTS ON ELECTRONIC SYSTEMS .....	4
1. Total Dose .....	4
2. Single Event Upsets (SEUs) .....	4
a. Soft Errors .....	5
b. Hard Errors .....	5
D. ERROR CORRECTION AND DETECTION (EDAC) .....	6
1. Theory .....	6
2. Implementation .....	7
3. Correcting Errors in Memory .....	8
III. MICROPROCESSOR INTERFACE .....	9
A. INTRODUCTION .....	9
B. SYSTEM CLOCK GENERATOR .....	9
C. BUS INTERFACE UNIT (BIU) .....	10
1. Bus Cycle .....	11
2. Address/Data Bus .....	12
3. Address Space .....	13
D. CHIP SELECT UNIT (CSU) .....	14
E. PIN TIMINGS .....	15
F. INTERRUPT CONTROL UNIT .....	18



IV. MEMORY BUS CONTROLLER DESIGN .....	19
A. DESIGN OVERVIEW .....	19
1. Constraints and Tradeoffs .....	19
2. System Block Diagram .....	21
3. System Operation .....	23
B. SYSTEM COMPONENT DESCRIPTION .....	24
1. CMOS Families Used .....	24
2. Mosaic MSM8256VLMB-25 256K x 8 CMOS SRAM .....	24
3. Harris ACS630MS EDAC Circuit .....	27
4. Transceivers, Latches, and Flip-flops .....	28
5. Glue Logic .....	30
C. SEQUENTIAL STATE MACHINE DESIGN .....	30
1. Actions Performed in Each State .....	33
2. State Machine Clock Rate .....	34
3. Synchronizing the State Machine to the Bus Cycle .....	36
4. State Machine Clock .....	36
5. Next State Logic .....	37
a. State Assignment .....	37
b. Master State Machine State Transition Table, Karnaugh Maps and State Diagram .....	38
c. Timing Verification and Chip-select Considerations .....	42
6. Control Signal Logic .....	43
a. General Information .....	43
b. General Timing Considerations .....	43
c. Memory Output Enable .....	44
d. Memory Write .....	45
e. EDAC Control .....	45
f. Memory Error Flags .....	47
g. Temporary Latch Enable .....	48
h. Data Bus Transceiver / Temporary Register Output Enables .....	48
i. Bus Transceiver Direction Control .....	50
7. Other Design Considerations .....	50
a. Bus Hold Time .....	50
b. System Power Up .....	50
V. MEMORY SYSTEM SOFTWARE REQUIREMENTS .....	53
A. SRAM INITIALIZATION .....	53
B. INTERRUPT TRIGGER LEVEL .....	53
C. INTERRUPT ACKNOWLEDGE / SINGLE-BIT ERROR SERVICE ROUTINE .....	53
D. DUAL-BIT ERROR INTERRUPT SERVICE ROUTINE .....	54
E. MEMORY WASH .....	54

VI. SYSTEM TESTING AND PERFORMANCE .....	55
A. BACKGROUND INFORMATION .....	55
B. DATA INTEGRITY TESTS .....	55
C. EDAC TEST .....	56
D. MEASUREMENT OF DUAL-BIT ERRORS IN UNINITIALIZED MEMORY .....	56
E. DATA BUS SIGNAL HOLD TIME MEASUREMENT .....	57
VII. RECOMMENDATIONS AND CONCLUSION .....	59
A. RECOMMENDATIONS .....	59
1. Short Term Recommendations .....	59
2. Long Term Recommendations .....	61
B. CONCLUSION .....	62
APPENDIX A. ESTIMATION OF UNCORRECTABLE MEMORY ERRORS ABOARD PANSAT .....	63
APPENDIX B. SYSTEM WAVEFORM DIAGRAMS .....	67
APPENDIX C. WIRE WRAP DEVELOPMENT BOARD SCHEMATICS .....	71
APPENDIX D. LIST OF NON-CRITICAL ADVANCED CMOS GATES .....	81
LIST OF REFERENCES .....	83
INITIAL DISTRIBUTION LIST .....	85

# **I. INTRODUCTION**

## **A. PURPOSE**

The purpose of this thesis is to document the design of a memory system that provides error detection and correction (EDAC) capabilities between an Intel 80C186XL microprocessor running at 7.3728 MHz and its addressable Static Random Access Memory (SRAM). This system was designed for use on board the Petite Amateur Navy Satellite (PANSAT). PANSAT is an experimental, low cost, lightweight, communications satellite that is currently being designed and built at the Naval Postgraduate School in Monterey, California.

## **B. SCOPE OF THESIS**

Chapter II provides background information including the PANSAT operational environment, radiation and its effect on electronic circuits, and an explanation of EDAC. In Chapter III, the Intel M80C186XL microprocessor, to be used aboard PANSAT, is discussed. The fourth chapter describes the memory system designed and Chapter V describes the software requirements of the memory system. Chapter VI discusses the testing performed on a wire wrap development board implementation of the system. Chapter VII presents recommendations for follow-on projects and the conclusion.



## **II. BACKGROUND INFORMATION**

This chapter provides background information so that the reader has an understanding of the PANSAT memory system requirements. Topics encompass microsatellite design philosophy, PANSAT mission life and operational environment, radiation effects on Complementary Metal-Oxide Semiconductor (CMOS) Integrated Circuits (ICs), and EDAC.

### **A. MICROSATELLITE DESIGN PHILOSOPHY**

Microsatellites are small sophisticated spacecraft, designed to perform limited missions in low earth orbits (LEOs). "By adopting a design philosophy which puts emphasis on electronics, rather than mechanical systems, and by taking a realistic 'cost-effective' approach to design, procurement and manufacture, it has been possible to construct spacecraft in time-scales and within budgets which have yet to be achieved by traditional aerospace industries." [Ref. 1] PANSAT is being designed to provide digital store and forward communications in the amateur frequency band. The noncritical nature of this mission relaxes the requirement for costly space grade components that would provide the highest level of reliability. Instead, PANSAT will use lower cost components in conjunction with redundancy and error correction methods to keep costs reasonable while providing protection against satellite system failure.

### **B. PANSAT MISSION LIFE AND OPERATING ENVIRONMENT**

The PANSAT Requirements Document specifies a two-year mission life and a LEO with an inclination between 28.5 and 90.0° [Ref. 2]. Current plans suggest that PANSAT will be launched during a Space Shuttle mission as a payload of opportunity. Because a specific shuttle mission has not been identified, the exact orbit of PANSAT is unknown at this time. However, the published Space Shuttle operational limits are altitudes between 203.7 km and 611.2 km and inclinations between 28.5° and 57.0° [Ref. 3].

## **1. Thermal Environment**

Thermal analysis has shown that the temperature inside PANSAT during operation will be between 0°C and 40°C [Ref. 4].

## **2. Radiation Environment**

In LEO, PANSAT will be partially protected from cosmic radiation by the combined effects of the solar wind and the earth's magnetic field [Ref. 5, p. 662]. The primary source of radiation in orbits of concern is low energy electrons (up to a few MeV) and relatively low energy protons (up to 40 MeV) encountered in the South Atlantic Anomaly (SAA). The SAA is an area encompassing approximately 45° latitude and 45° longitude centered near 20° N, 20° W where the Van Allen radiation belts are closest to the earth. Another source of radiation in LEO is occasional energetic protons (up to 100 MeV) associated with solar flare activity [Ref. 5, p. 712]. PANSAT should experience a radiation dose (called ionizing dose) rate of approximately one krad (Si) per year [Ref. 5, p. 452].

# **C. RADIATION EFFECTS ON ELECTRONIC SYSTEMS**

Radiation causes two types of effects in electronic systems: an effect from total dose and an effect from the dose rate that manifests itself as a single event upset (SEU).

## **1. Total Dose**

Total dose effects are a result of the cumulative amount of radiation the system has absorbed. After an electronic system has absorbed a critical amount of radiation, performance begins to degrade until finally the system fails. The failed condition is called latchup. Bulk CMOS ICs begin to degrade at an ionizing dose of approximately two krad (Si) and latchup begins at approximately 100 krad (Si) [Ref 5,p. 640]. Because of its short operational lifetime, total dose radiation is not expected to pose a problem for the CMOS circuitry on board PANSAT.

## **2. Single Event Upsets (SEUs)**

Single Event Upsets (SEUs) are a result of the rate of individual strikes on the IC by high energy particles resulting in bit errors. SEUs primarily affect large-scale integration

(LSI) and very large-scale integration (VLSI) ICs [Ref. 5, p. 416]. The number of upset bits is proportional to the energy of the particles. A study suggests that multiple upset bits in the same byte of SRAM devices are infrequent. In the two SRAM devices studied, one exhibited no double errors per byte and the other exhibited double errors per byte only when irradiated with high energy ions [Ref. 6, p. 1509].

#### ***a. Soft Errors***

The most prevalent type of SEU is called a "soft error." Usually within tens of nanoseconds after particle penetration the device recovers with no permanent damage [Ref. 5, p. 416]. The bit in error is then corrected when it is overwritten by a new datum. CMOS SRAM devices aboard the UoSAT-2 microsatellite in a polar LEO exhibited soft SEU error rates in the range of  $10^{-6}$  errors per bit per day. [Ref. 7, p. 2342].

To reduce the SEU soft error effects on the system, small integration (SSI) and medium scale integration (MSI) CMOS ICs are used in the memory bus controller. Because the SRAM (a VLSI device) is sensitive to SEUs, EDAC is incorporated into the microprocessor addressable SRAM. Non-radiation hardened SRAM together with EDAC offers PANSAT three critical advantages over radiation-hardened memory: lower cost; greater availability; and higher package densities [Ref. 6, p. 1507].

#### ***b. Hard Errors***

SEUs also include single event latchup (SEL) or "hard errors." SELs are characterized by stuck bits; i.e., the bit in error cannot be overwritten by a new datum. If the transistors that comprise the bit in error are not burned out, SELs can usually be cleared by powering down the effected IC [Ref. 5, p. 483]. SELs occur less frequently than soft SEUs. Experimentation suggests that SELs are caused by higher energy particles [Ref. 7, p. 1509]. Bulk CMOS ICs in geosynchronous (high earth) orbits have exhibited SEL rates between  $10^{-6}$  and  $10^{-7}$  errors per bit per day [Ref. 5, p. 665]. Microsatellites in LEOs have used unhardened CMOS SRAM successfully without design consideration for SELs. Furthermore, EDAC will delay the adverse effects of SELs. Ignoring soft errors, EDAC will correct one stuck bit at every memory address. When accumulated stuck bits cause more than one error at a memory address, the address becomes unusable.

Due to the critical nature of the microprocessor, memory and other digital control system (DCS) components, PANSAT has two identical DCSs. One DCS system is

powered down while the other system is operating. If uncorrectable errors are observed, control may be switched to the other DCS system. Accumulated SELs in the first DCS should be cleared so if uncorrectable errors again present a problem control can be switched back.

## **D. ERROR CORRECTION AND DETECTION (EDAC)**

### **1. Theory**

Error detection and correction is accomplished using a Hamming code to generate a check word for each data word stored in memory. The level of EDAC protection needed determines the number of check word bits per data word. Providing single-bit error correction with dual-bit error detection to eight data bits requires five check bits; to provide the same level of protection to 16 data bits requires six check bits. To determine the number check word bits, the expected error rate should be balanced against the costs of the additional memory required to store the check bits. Single-bit error correction with dual-bit error detection is common and will provide adequate protection for the error rates expected on board PANSAT. Single-bit error correction implies that any single-bit error in the data word *or* check word is correctable. Likewise, dual-bit error detection implies that all dual-bit errors in the data word *or* check word are detectable. Three or more bit errors may not be detected or may be interpreted as a single-bit or dual-bit error.

When a data word is stored in memory, the associated check word is also stored. During a memory read operation the data word and corresponding check word are retrieved from memory. A new check word based on the data word from memory is generated and compared to the check word retrieved from memory. If the two check words are identical the data word is assumed correct. When the check words are not the same an error has been detected. The total number of errors in both the data word and check word from memory is determined by the number of miscompared bits in the two check words. Correctable errors are identified and corrected by inverting the bit(s) in error. Words having detectable-only errors are flagged and the corrupt data is discarded. After detecting a correctable error an



EDAC circuit may generate a syndrome code. A syndrome code identifies the bit(s) in error. The syndrome code can be used to identify bad memory cells.

## **2. Implementation**

Commercially available single-bit error correction with dual-bit error detection EDAC ICs are designed to generate check words from at least 16-bit data words. These devices may be used with fewer data bits by tying the unused data and check bit I/O pins to appropriate values through pull up and pull down resistors. When used as a 16-bit device, byte (eight-bit) data writes to memory cannot be accomplished directly because modifying one byte of a word requires modifying the check word. Both bytes of the word are needed to generate the check word. There are two methods to manage this situation. The first is to avoid it by using two EDAC ICs, one for each byte of the data word. Each byte then has its own check word; thus either byte may be written without affecting the other. The second method is to use a "read-modify-write" scheme to make byte writes from the source (usually the microprocessor) appear as word writes to memory. This is accomplished by performing a "pre-read" of the word that contains the byte to be overwritten and storing it in a temporary register. Then a new data word is created by combining the new byte from the source with the appropriate byte from the temporary register. A new check word is generated based on this new data word and both words are stored in memory.

The two EDAC circuit method provides better error correction capabilities than the "read-modify-write" method. The penalty for this improved correction capability is the additional memory required to store two five-bit check words compared to the memory required to store one six-bit check word. Each SRAM device is eight bits wide. Storing ten check bits would require two additional memory devices for each pair of devices used to store the data words. Storing six check bits requires only one additional memory device for each pair of memory devices used to store the data words. Additionally, the EDAC and the SRAM are the two most expensive ICs in this design. The improvement in error handling capabilities do not offset the penalties of increased Printed Circuit Board (PCB) space and increased cost. Therefore this design uses the "read-modify-write" technique to perform byte write operations.

### **3. Correcting Errors in Memory**

An advantage to the "read-modify-write" method is that single bit errors can be corrected in memory during the read bus cycle in which they are detected. Errors are corrected during the pre-read operation and then the corrected data is stored in the temporary register. A new check word is generated and stored in memory with the corrected data while the corrected data is passed to the microprocessor. In a system without a temporary data register the corrected data is passed to the microprocessor but the uncorrected data remains in memory until it is overwritten. The technique used in the UoSAT series microsattellites is to leave the corrupt data in memory until it is overwritten during the "memory wash" operation.

The memory wash is a continuous background operation performed to prevent errors from accumulating in memory. This operation involves cycling through the entire SRAM space correcting all errors. Without the temporary data register the wash operation requires reading a data word from memory into a temporary location internal to the microprocessor. If an error is detected, the microprocessor is alerted via an interrupt signal and the corrected data word is then written back to memory at the same address from which it was read. Otherwise the data word is discarded and the process is repeated with the next memory address. Because an external temporary data register allows errors to be corrected during the same bus cycle in which they are detected, a wash of one data word always requires only one read bus cycle.

Using an EDAC circuit in the SRAM system on board PANSAT allows the use of non radiation hardened memory devices without sacrificing reliability. The following chapters document the design of a memory system that provides single-bit error detection and correction in a single SRAM bus cycle without adding wait states to the bus cycle.

### III. MICROPROCESSOR INTERFACE

#### A. INTRODUCTION

The primary component of the PANSAT Digital Control System (DCS) is the Intel M80C186XL-10 16-Bit Microprocessor. The M80C186 is available in several versions and each version is available with different input clock rates. Throughout this thesis, the term M80C186 will refer to any version of the microprocessor and the term M80C186XL will refer specifically to the XL-10 version to be used on board PANSAT.

The M80C186 provides a high level of integration making it ideal for use on board PANSAT. The M80C186 microprocessor incorporates in one package the M80C86 Central Processing Unit (CPU) with the following features:

- a clock generator
- a two channel Direct Memory Access (DMA) controller
- a programmable interrupt controller
- a programmable chip-select controller
- a programmable wait state generator

This chapter discusses these features in the context of designing a memory system with EDAC for the M80C186XL. Typically logic signals and device pin names that are asserted low are labeled with a bar over the title. This thesis will identify these same logic signals and device pin names by preceding the signal or pin name with a slash (/).

#### B. SYSTEM CLOCK GENERATOR

The maximum CPU clock rate of the M80C186XL used in the PANSAT DCS system is 20 MHz. The requirement to interface with the other DCS components limits the CPU clock rate to 7.3728 MHz. The M80C186 internal clock generator performs a frequency divide-by-two operation to a 14.7456 MHz external clock signal applied to the X1 pin. The input clock signal is designated CLKIN. The CLKOUT output pin on the M80C186 provides a 50% duty cycle waveform at the CPU clock rate. The M80C186XL maximum skew from CLKIN to CLKOUT (designated  $T_{CICO}$ ) is 25 ns. The M80C186XL CLKOUT signal has sufficient capability to drive a 100 pF load [Ref. 8, p. D-4]. All M80C186 pin timings are specified relative to the CLKOUT signal.

The clock generator also provides a system reset output signal (RESET). During power-up the /RES input pin is held low by a Resistor-Capacitor (RC) circuit while the

external clock and power are delivered to the microprocessor. The M80C186 pins assume their reset state a maximum of 28 periods after X1 and V<sub>CC</sub> stabilize. The RESET output pin goes low within 2½ CLKOUT cycles after /RESET goes high. The first bus activity begins seven CLKOUT cycles after /RES goes high.

### C. BUS INTERFACE UNIT (BIU)

The M80C186 BIU controls the flow of data between the microprocessor, memory, and integrated peripheral units. The BIU generates the address, bus control and status signals for CPU initiated bus cycles. During DMA bus cycles the BIU provides bus control and status signals and the M80C186 integrated DMA controller generates the address. Table 1 lists the bus control signals.

Symbol	Signal Name	Function	Reset State
/BHE	Bus High Enable	Enables data on most significant half of data bus.	Float
ALE	Address Latch Enable	Enables external address latches.	Low
/WR	Write Strobe	Indicates data on bus ready to be written.	High for one CLKOUT cycle, then float.
/RD	Read Strobe	Indicates M80C186 is performing a read cycle.	High for one CLKOUT cycle, then float.
/S6:0	Bus Cycle Status	Indicates bus cycle type.	High for one CLKOUT cycle, then float.
DT-/R	Data Transmit/Receive	Controls the direction of flow through the data bus transceivers	Float
/DEN	Data Enable	Enables the data bus transceivers.	High for one CLKOUT cycle, then float.

Table 1 Bus Status and Control Output Signals

## 1. Bus Cycle

The M80C186 bus cycle consists of multiple CPU clock cycles. These clock cycles are designated "T-states." A normal bus cycle consists of four T-states identified as T-1, T-2, T-3 and T-4. The T-state corresponding to an idle bus is identified as T-I. The T-states are separated by the falling edge of CLKOUT. Each T-state is divided into two phases. Phase one corresponds to a low CLKOUT signal; phase two corresponds to a high CLKOUT signal. Throughout this thesis a specific phase of the bus cycle is identified by appending the phase number to the T-state designation, e.g., T-1-1 (see Figure 1).

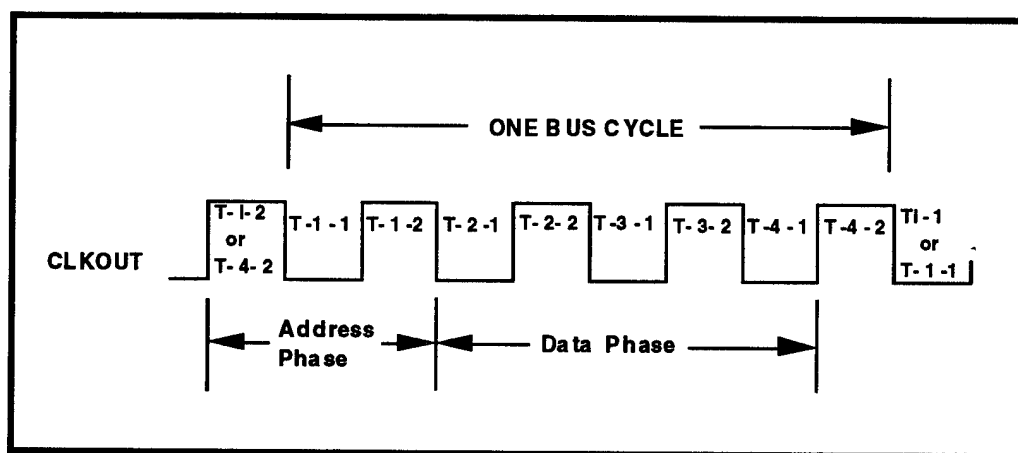


Figure 1 Standard Bus Cycle

Software and/or hardware controls the insertion of wait states. The M80C186 may be programmed to insert wait states into the bus cycle when a specific memory device or peripheral is addressed. Additionally, the two bus ready input pins, ARDY and SRDY, may be used to signal the BIU that the peripheral or memory addressed is ready to complete the bus cycle. The BIU inserts wait states after T-3, maintaining the same control signal outputs as in T-3. After the ready conditions are met, the BIU allows the bus cycle to continue into T-4 and complete. To avoid inserting wait states when using hardware control, a logic signal signifying the decision must be available by the Synchronous Ready Transition Setup Time ( $T_{\text{SRVCL}}$ ) before T-3-1. M80C186XL  $T_{\text{SRVCL}}$  is 20 ns.

There are several types of bus cycles. The Bus Cycle Status pins (S2:0) provide an external indication of the current bus cycle type (see Table 2). S2:0 are valid T-1-1 through T-3-1.

/S2	/S1	/S0	Bus Cycle
0	0	0	Interrupt Acknowledge
0	0	1	Read I/O
0	1	0	Write I/O
0	1	1	Halt
1	0	0	Instruction Fetch
1	0	1	Read Data from Memory
1	1	0	Write Data to Memory
1	1	1	Passive (no bus cycle)

Table 2 Bus Cycle Status Information

## 2. Address/Data Bus

The M80C186 uses a time multiplexed bus to transfer address and data. The bus cycle is divided into two phases: the address phase and the data phase. Addressing information is transferred over the 20 pins designated as A19:1 and AD15:0. The address phase begins during phase two of the last T-state prior to the start of a bus cycle and extends through T-1-2. The address is then latched in external latches controlled by the Address Latch Enable (ALE) signal. Data is transferred over the 16 pins designated as AD15:0 during the data phase. The data phase of a bus cycle begins in T-2-1 and extends through T-4-1 (see Figure 1).

### 3. Address Space

With 20 address bits, the M80C186 can directly address up to 1 MB of memory. When used with a 16-bit data bus the memory is divided into two banks, each one byte wide, of up to 512K. Address bits A19:A1 provide the same address to both memory banks. Then A0 and /BHE are used to select either one or both of the addressed bytes (see Figure 2 and Table 3).

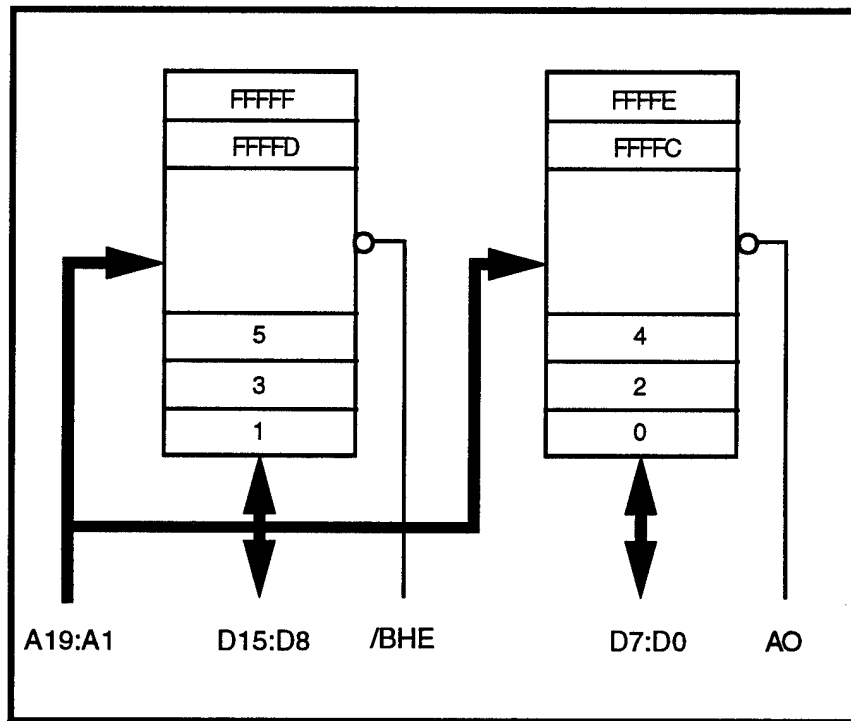


Figure 2 Implementation of 16-Bit Data Bus

<b>/BHE</b>	<b>A 0</b>	<b>Transfer Type</b>
0	0	Word (D15:0)
0	1	High Byte (D15:8)
1	0	Low Byte (D7:0)
1	1	Refresh (not used)

Table 3 /BHE and A0 Encoding

The M80C186 performs both byte and word transfers on the data bus. To simplify operation, the control signals issued by the BIU during byte read operations are identical to the control signal issued during word read operations. The M80C186 simply floats both halves of data bus during T-2-2 through T-4-2 and ignores the unaddressed byte of data.

#### D. CHIP SELECT UNIT (CSU)

The M80C186 internal CSU decodes the bus cycle address to generate a chip-select signal. The CSU asserts 14 control pins to select a specific memory or peripheral device. One chip-select signal goes active in T-1-1 when the bus cycle address is within software specified range. The chip-select signals remain active until T-4-2. They are typically programmed as follows:

- /UCS—Mapped only to upper memory address space and selects the BOOT memory device. The upper limit always ends at address 0FFFFFFH. The lower limit is programmable for maximum block size of 256K. [Ref. 9, p. 9-21]
- /LCS—Used to select the devices corresponding to the lowest memory address space. The lower limit is address 0H and the upper limit is programmable for maximum block size of 256K. [Ref. 8, p. 6-2]
- /MCS3:0—Used to select a block of memory address space from 8K to 512K. Each chip-select pin is active for one fourth of the block size. The starting address of the block is programmable but must be an integer multiple of the block size (0H is a multiple of every block size). This prevents /MCS3:0 from selecting the entire address space between /LCS and /UCS. [Ref. 8, p. 6-3]



- /PCS7:0—Used to select memory or I/O address space and select peripheral devices or generate a DMA acknowledge strobe. Each chip-select goes active for 128 bytes of the 896 byte block. The /PCS block start address can begin on any 1K boundary. [Ref. 8, pp. 6-2-6-3]

## E. PIN TIMINGS

Figures 3, 4, and 5 illustrate the pin waveforms and define the pin timing parameters. Table 4 lists the pin timing parameter limits pertinent to the design of the memory bus controller. All timing specifications given are for operation in the  $-55^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$  range with  $C_L = 50\text{--}100\text{pF}$  (except CLKOUT where  $C_L = 50\text{--}200\text{pF}$ ).

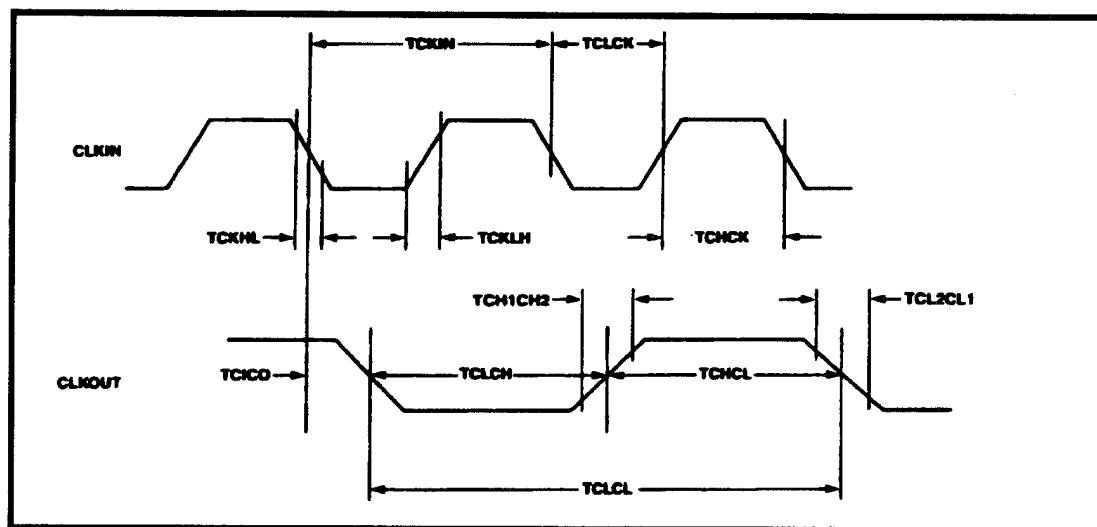
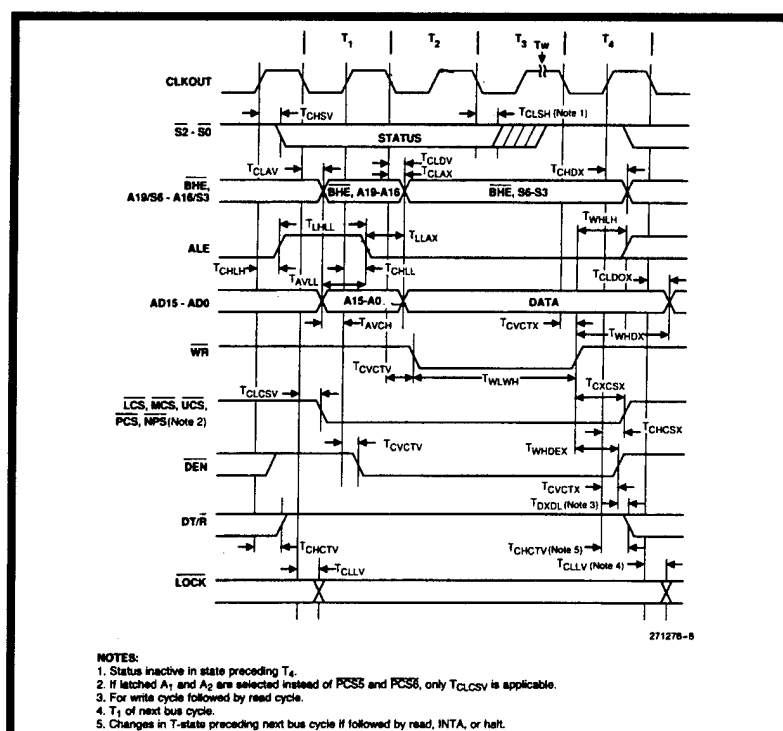
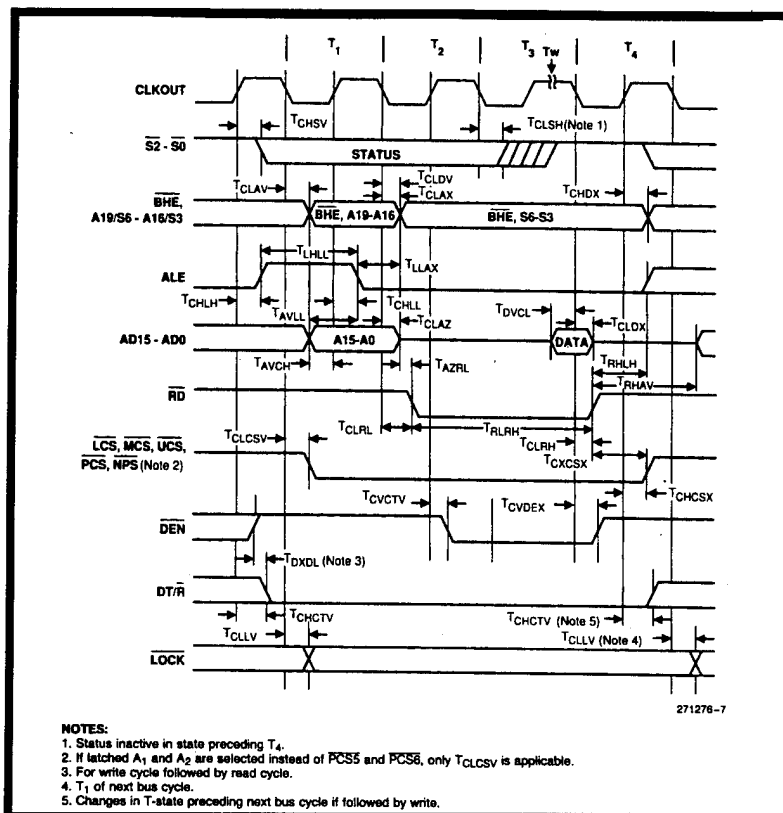


Figure 3 M80186 CLKIN-CLKOUT Relationship



Symbol	Parameter	Min (ns)	Max (ns)
T <sub>DVCL</sub>	Data In Setup	20	
T <sub>CLDX</sub>	Data In Hold	5	
T <sub>SRYCL</sub>	Synchronous Ready Transition Setup Time	20	
T <sub>CLAV</sub>	Address Valid Delay	3	44
T <sub>CHLH</sub>	ALE Active Delay		30
T <sub>CHLL</sub>	ALE Inactive Delay		30
T <sub>CLDV</sub>	Data Valid Delay	3	40
T <sub>CLDOX</sub>	Data Hold Time	3	0
T <sub>CVCTV</sub>	Control Active Delay 1	3	56
T <sub>CHCTV</sub>	Control Active Delay 2	3	44
T <sub>CVCTX</sub>	Control Inactive Delay	4	44
T <sub>CVDEX</sub>	/DEN Inactive Delay	3	44
T <sub>CLRL</sub>	/RD Active Delay	3	44
T <sub>CLRH</sub>	/RD Inactive Delay	3	44
T <sub>CHSV</sub>	Status Active Delay	3	45
T <sub>CLSH</sub>	Status Inactive Delay	3	46
T <sub>CLCSV</sub>	Chip-Select Active Delay		45
T <sub>CHCSX</sub>	Chip-Select Inactive Delay	3	35
T <sub>CICO</sub>	CLKIN to CLKOUT Skew		25
T <sub>CLCH</sub>	CLKOUT Low Time (min)	$0.5 \times T_{CLCL} - 6$	
T <sub>CHCL</sub>	CLKOUT High Time (min)	$0.5 \times T_{CLCL} - 6$	
T <sub>CH1CH2</sub>	CLKOUT Rise Time		10
T <sub>CL2CL1</sub>	CLKOUT Fall Time		10

Table 4 M80C186XL Pin Timings

## **F. INTERRUPT CONTROL UNIT**

The M80C186 internal interrupt control unit processes all maskable interrupts when used in Master Mode (as on board PANSAT). “The Interrupt Control Unit synchronizes and prioritizes interrupt sources and provides the interrupt vector to the CPU.” [Ref. 8, p. 8-1]

Besides five internal interrupt sources, there are four external interrupt pins (INT3:0). During processor boot or system reset all interrupts are masked off, each interrupt is assigned a default priority level, and external interrupts are set to Edge-trigger Mode. Independently each interrupt signal may be unmasked and have its priority and trigger mode changed through software control. There are two interrupt trigger modes: edge-triggered and level-triggered. The trigger mode determines when an interrupt signal is recognized by the interrupt control unit. Edge-triggered interrupts are recognized on the rising edge of the interrupt signal. After an edge-triggered interrupt is serviced the interrupt control unit will not recognize another interrupt from that same source until the interrupt signal goes low and then high to present another rising edge. A level-triggered interrupt is recognized whenever the interrupt pin is high. If the interrupt pin remains high after the interrupt is serviced the interrupt control unit will present another interrupt request to the CPU.

## **IV. MEMORY BUS CONTROLLER DESIGN**

### **A. DESIGN OVERVIEW**

This section contains an overview of the memory bus controller design. Design constraints and tradeoffs are discussed and a system block diagram is provided. The action performed by the bus controller in each T-state of an SRAM access bus cycle is provided in tabular format.

#### **1. Constraints and Tradeoffs**

The design objective was to build a memory bus controller using readily available components. Five major constraints influenced the design:

- Suitability for use in a short duration LEO radiation environment
- PCB area required
- Speed of operation
- Power required
- Dollar cost

The small size of PANSAT limits the area available for the memory system. The goal is to fit the entire DCS consisting of a M80C186XL, EDAC circuitry, 512K of SRAM, 256K of Read-Only Memory (ROM), an eight-bit peripheral bus controller, and an analog-to-digital converter on a 5.5" by 8.5" PCB.

The SRAM bus controller should not be the DCS subsystem that limits the CPU clock speed. At this point in the PANSAT design process, other DCS subsystems have limited the CPU clock speed to 7.3728 MHz. A goal of this design was to not force wait states into every bus cycle accessing SRAM. The error detection process was to be transparent to the microprocessor.

Dependent upon solar panels to provide power, PANSAT will operate on a tight power budget. Generally, faster systems require more power. Additionally, radiation hardened components usually require more power than their non-hardened counterparts. Beyond the cost in terms of power required, the dollar cost of using radiation hardened parts must be considered. A radiation hardened part, if available, costs approximately 20 times what the comparable high reliability, non hardened part costs. In view of the relatively benign radiation environment in which PANSAT will operate, it is not cost effective to use radiation hardened parts.

Evaluating these constraints led to the following design decisions:

- CMOS was chosen over other logic families due to its low power requirement.
- Standard SSI and MSI IC parts were chosen because they are readily available and offer decreased susceptibility to SEUs at a reasonable dollar cost, although at high cost in terms of PCB area.

Considering the low power requirements of CMOS ICs running at the low clock rates required by this design, power was judged a less critical consideration than system size. To limit the size and obtain the required speed, decisions made by the bus controller were minimized. A read-and-correct operation into a temporary register is performed during the first part of every SRAM bus cycle. This operation is not required during word write or “no error” read SRAM accesses. Furthermore, during all SRAM read accesses, the data word from memory is written back to memory from the temporary register while it is sent to the microprocessor regardless of whether it has been changed during the correction process. These extra operations consume power. Additionally, during a word write, a dual-bit error that would have otherwise been overwritten may be detected and cause the microprocessor to take unnecessary action. However, the additional logic required to identify and eliminate these conditions would require more components and force the bus controller to run slower. Considering the number of dual-bit errors expected and the low power required to perform these extra operations, these tradeoffs are acceptable.

## 2. System Block Diagram

Table 5 describes each control signal used by the memory system. Figure 6 provides a block diagram of the system.

Control Signal	Symbol	Source	Function
Bus High Enable	/BHE	Microprocessor	Signals memory access requires high data byte.
Latched Bus High Enable	/LBHE	Address Latch	Signals memory access requires high data byte.
Latched A0	LA0	Address Latch	Signals memory access requires low data byte.
Memory Chip Select	/CS	Microprocessor	Enables memory device addressed. Identifies SRAM access bus cycle.
Address Latch Enable	ALE	Microprocessor	Controls address latch. Identifies beginning of bus cycle.
Clock Out	CLKOUT	Microprocessor	Provides clocks for synchronous bus control.
Reset	RESET	Microprocessor	Signals system reset.
Interrupt Acknowledge	/ERR_ACK	Microprocessor	Acknowledges interrupt from bus controller.
Direction	DT-/R	Microprocessor	Controls direction of data flow through data bus transceivers.
High Transceiver Enable	/XVER_HI_EN	Bus Controller	Enables the high data byte bus transceiver.
Low Transceiver Enable	/XVER_LO_EN	Bus Controller	Enables the low data byte bus transceiver.
Temporary Latch Enable	/TEMP_LE	Bus Controller	Latches data in the temporary register.
High Temporary Latch Output Enable	/TEMP_HI_OE	Bus Controller	Enables the output of the high byte of the temporary register.
Low Temporary Latch Output Enable	/TEMP_LO_OE	Bus Controller	Enable the output of the low byte of the temporary register.
Memory Write	/MEM_WR	Bus Controller	Latches data into memory.
Memory Output Enable	/MEM_OE	Bus Controller	Enables the memory output.
EDAC Control	S1, S0	Bus Controller	Controls state of EDAC IC
Error Flags	SEF, DEF	EDAC Circuit	Signals detection of correctable (SEF) or uncorrectable (DEF) errors

Table 5 System Control Signals

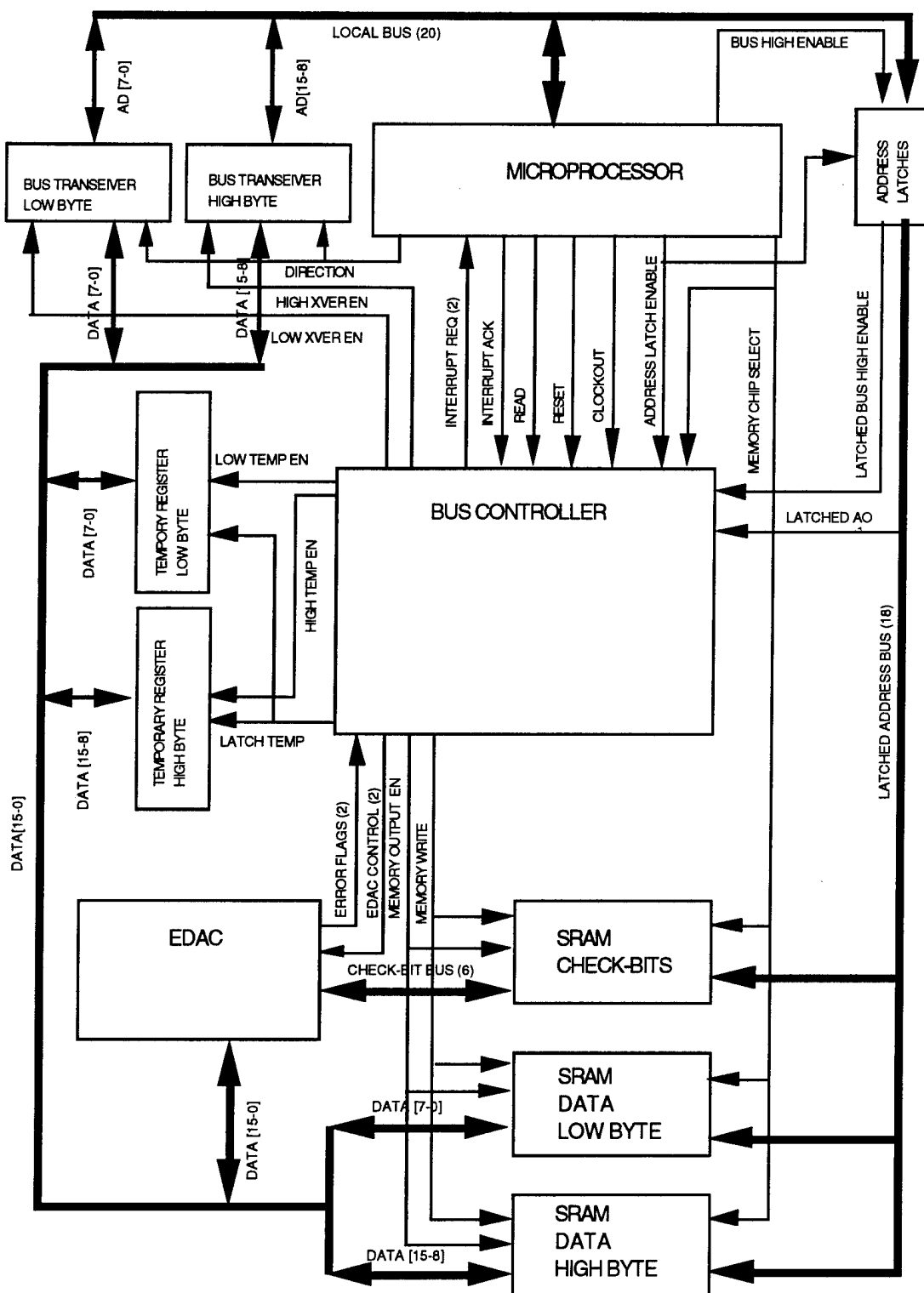


Figure 6 System Block Diagram



### 3. System Operation

Table 6 lists the action taken by the bus controller in each T-state phase of an SRAM access bus cycle. Table 7 shows the state of the bus transceiver and temporary register outputs during T-3-2 through T-4-1.

<b>T-State Phase</b>	<b>Bus Controller Action</b>
T-I-1, T-I-2, T-4-2	Idle: EDAC circuit set for memory read; Transceiver and Temporary Register outputs disabled.
T-1-1	Enable SRAM output.
T-1-2	No operation
T-2-1	Latch data and check words in EDAC circuit; Disable SRAM output.
T-2-2	Latch DEF and SEF; Correct data in EDAC circuit and put corrected data on bus.
T-3-1	Latch corrected data into temporary register; Set EDAC circuit to generate check word.
T-3-2	Enable bus transceiver and temporary register outputs as required for bus cycle type. (See Table 7.) Generate check word.
T-4-1	Latch data and check words into SRAM.

Table 6 Bus Controller Action by T-State

<b>Type Memory Access</b>	<b>High Byte Transceiver Output</b>	<b>Low Byte Transceiver Output</b>	<b>High Byte Temporary Register Output</b>	<b>Low Byte Temporary Register Output</b>
READ (word or byte)	Enabled	Enabled	Enabled	Enabled
WRITE Low Byte	Disabled	Enabled	Enabled	Disabled
WRITE High Byte	Enabled	Disabled	Disabled	Enabled
WRITE Word	Enabled	Enabled	Disabled	Disabled

Table 7 Transceiver/Temporary Register Outputs T-3-2 through T-4-1

## B. SYSTEM COMPONENT DESCRIPTION

### 1. CMOS Families Used

Two families of CMOS ICs are used in this design, High-speed CMOS and Advanced CMOS. An Advanced CMOS (designated AC) device is approximately three to five times faster than the equivalent High-speed CMOS (HC) device, but it consumes approximately 50% more power. AC devices are more expensive and harder to obtain than HC devices. For these reasons, AC components are used in this design only where required to attain the desired speed. However, because the “glue logic” devices used have more than one gate per package some AC gates are used in non speed critical logic circuits to reduce unused gates and minimize the total number of packages required. A list of the AC gates used in this design that may be replaced by HC gates without affecting system performance is provided in Appendix E. These gates should be used in other speed critical logic circuits on the DCS PCB.

High reliability, military temperature rated components are used in this design. All the devices except the SRAM are Ceramic Dual-In-line Packages (CERDIP) processed to MIL-STD-833 specifications. The SRAM devices are ceramic high density Vertical-In-Line (VIL™) packages processed to non-compliant MIL-STD-833 Method 5004.

### 2. Mosaic MSM8256VLMB-25 256K x 8 CMOS SRAM

Mosaic MSM8256VLMB-25 256K x 8 CMOS SRAM devices were selected for this design for their fast access time (25 ns) and space saving VIL™ packaging (see Figure 7). These SRAM devices have three control inputs; Chip-select, Write Enable, and Output Enable. Figures 8 and 9 and Table 8 provide the SRAM timing specifications.

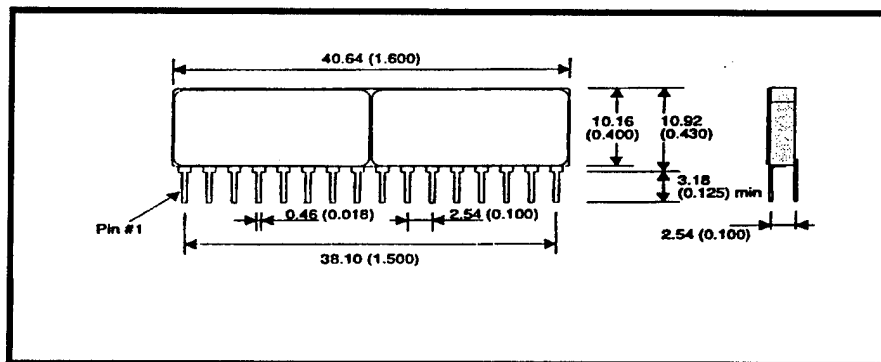


Figure 7 Mosaic MSM8256 256K x 8 SRAM VIL™ Package

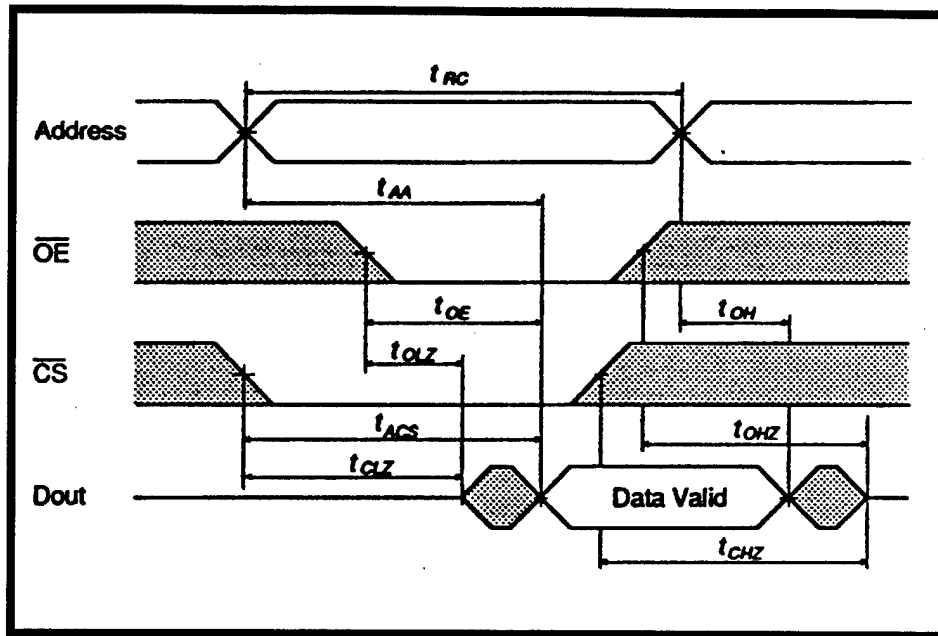


Figure 8 SRAM Read Cycle Waveforms

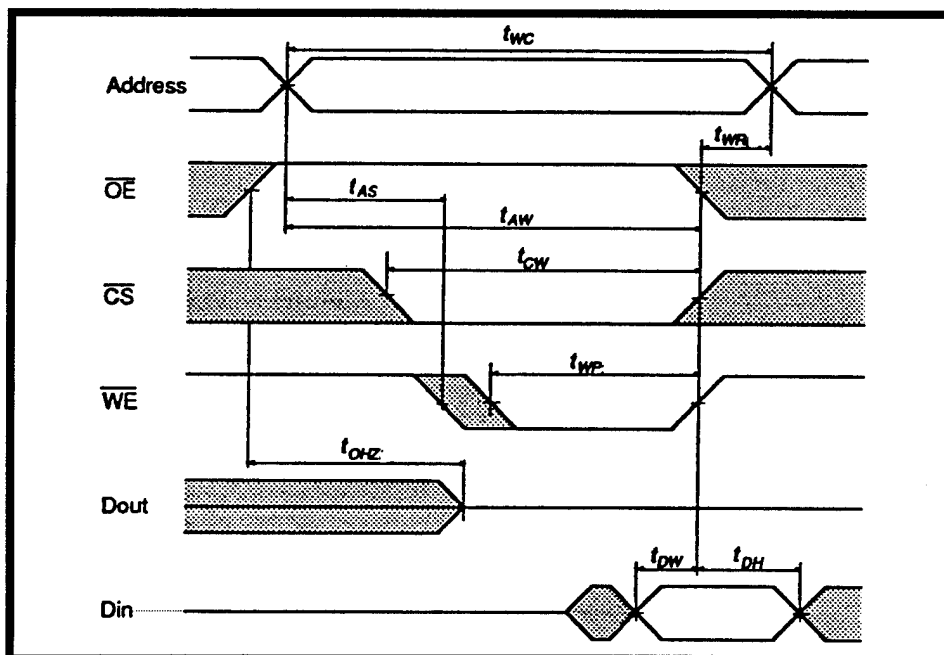


Figure 9 SRAM Write Cycle Waveforms

Parameter	Symbol	Min (ns)	Max (ns)
Read Cycle Time	$t_{RC}$	25	
Address Access Time	$t_{AA}$		25
Chip Select Access Time	$t_{ACS}$		25
Output Enable to Output Valid	$t_{OE}$		25
Output Hold from Address Change	$t_{OH}$	3	
Chip Select to Output Enabled	$t_{CLZ}$	5	
Output Enable to Output Enabled	$t_{OLZ}$	0	
Chip Deselected to Output Disabled	$t_{CHZ}$	0	10
Output Disable to Output Disabled	$t_{OHZ}$	0	10
Write Cycle Time	$t_{WC}$	25	
Chip Selected to End of Write	$t_{CW}$	17	
Address Valid to End of Write	$t_{AW}$	20	
Address Setup Time	$t_{AS}$	0	
Write Pulse Width	$t_{WP}$	17	
Write Recovery Time	$t_{WR}$	3	
Data to Write Time Overlap	$t_{DW}$	15	
Data Hold from Write Time	$t_{DH}$	0	
Output Disable to Output Disabled	$t_{OHZ}$	0	10
Output Active from End of Write	$t_{OW}$	0	

Table 8 MSM8256-25 SRAM Timing Specifications

### 3. Harris ACS630MS EDAC Circuit

The Harris ACS630MS EDAC Circuit was selected for its simple four state operation and fast processing time. This circuit requires two control inputs, designated S0 and S1, and has a 16-bit port for data word I/O and a six-bit port for check word I/O and syndrome output (see Figure 10). Table 9 details the circuit operation and Table 10 provides timing specifications for the ACS630MS.

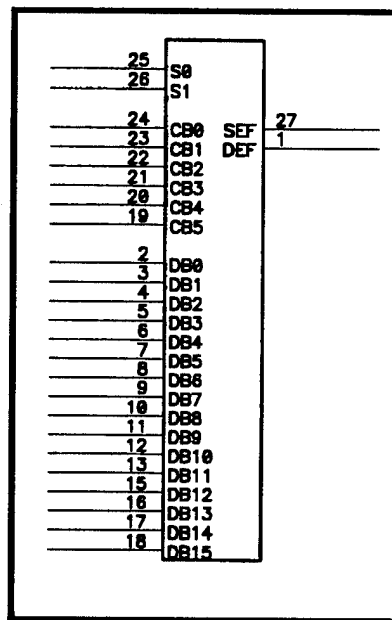


Figure 10 Harris ACS630

Control		EDAC Function	Data I/O	Check Word I/O	Error Flags	
S 1	S 0				SEF	DEF
Low	Low	Generate Check Word	Input data	Output Check	Low	Low
Low	High	Read Data and Check Word	Input data	Input Check Word	Low	Low
High	High	Latch and Flag Error	Latch data	Latch Check Word	Enabled	Enabled
High	Low	Correct Data Word and Generate Syndrome	Corrected Data	Syndrome Bits	Enabled	Enabled

Table 9 Harris ACS630MS Control Functions

Parameter	Description	Max (ns)
Propagation Delay	Data Word to Check Word ( <b>Generate</b> )	24.5
Setup Time	Input to S1	10
Hold Time	S1 to Input Removed	5
Propagation Delay	S1 to Error Flag (SEF or DEF) ( <b>Latch</b> )	15.5
Propagation Delay	S0 to Output Valid ( <b>Correct</b> )	20.5
Propagation Delay	S0 to Outputs Disabled ( <b>Read</b> )	21.5

Table 10 Harris ASC630MS Timing Specifications ( $V_{CC}=5V$ ,  $-55^{\circ}$  to  $+125^{\circ}C$ )

#### 4. Transceivers, Latches, and Flip-flops

Harris CD54AC245/3A Three-state Octal Bus Transceivers are used to isolate the SRAM data bus from the microprocessor local bus. They were selected for their low propagation and fast output enable times.

Two Harris CD54AC573/3A Three-state Octal Transparent Latches are used as the temporary register. These latches were selected because of their low output enable time and their low data setup and hold time requirements. Three Harris CD54HC573/3A are used to latch the address from the local bus during the address phase of the bus cycle. The

transparent feature allows the address to flow through the latch to the SRAM without first latching it, thus expediting availability of the address to memory.

A Harris CD54AC74/3A Dual D-type Flip-flop with Set and Reset provides two state bits for the bus controller state machine. This device was chosen for its fast transition time and low setup and hold timing requirements.

A Harris CD54AC374/3A Three-state Octal D-type Flip-flop provides two bits for the bus controller state machine. Two flip-flops latch the error flags from the EDAC circuit and the four remaining flip-flops latch bus controller output signals. This device was chosen also for its fast transition time and low setup and hold timing requirements.

Table 11 provides the pertinent timing specifications for these devices.

Parameter	AC74		AC245		AC374		AC573		HC573		Units
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
Propagation Delay	2.6	9.1	2.1	7.7	2.9	9.8	2.2	7.7	2.2	38	ns
Enable Time	N/A	N/A	3.7	12.7	3.7	13.2	2.7	9.5	2.7	33	ns
Disable Time	N/A	N/A	3.7	12.7	3.7	13.2	3	10.5	3	33	ns
Data Setup Time	4.3	-	N/A	-	2	-	2	-	13	-	ns
Data Hold Time	0	-	N/A	-	2	-	2.6	-	10	-	ns

Table 11 Harris Device Timing Parameters ( $V_{CC}=5V$ ,  $-40^{\circ}$  to  $+80^{\circ}C$ )

## 5. Glue Logic

This design incorporates the following Harris logic devices:

- CD54AC04/3A Hex Inverter
- CD54HC04/3A Hex Inverter
- CD54AC08/3A Quad Two-input AND Gate
- CD54HC08/3A Quad Two-input AND Gate
- CD54AC32/3A Quad Two-input OR Gate
- CD54HC32/3A Quad Two-input OR Gate

Table 12 provides the propagation delays for these devices.

Part	Min Propagation Delay (ns)	Max Propagation Delay (ns)
AC04	1.7	5.9
HC04	1.7	21
AC08	2.2	7.9
HC08	2.2	23
AC32	2.4	8.6
HC32	2.4	23

Table 12 Harris Glue Logic Propagation Delays ( $V_{CC}=5V$ ,  $-40^{\circ}$  to  $+85^{\circ}C$ )

## C. SEQUENTIAL STATE MACHINE DESIGN

A sequential state machine is used to generate the required control signals for the SRAM system. The schematic for this state machine is shown in Figures 11 and 12. This section describes the design of this state machine.



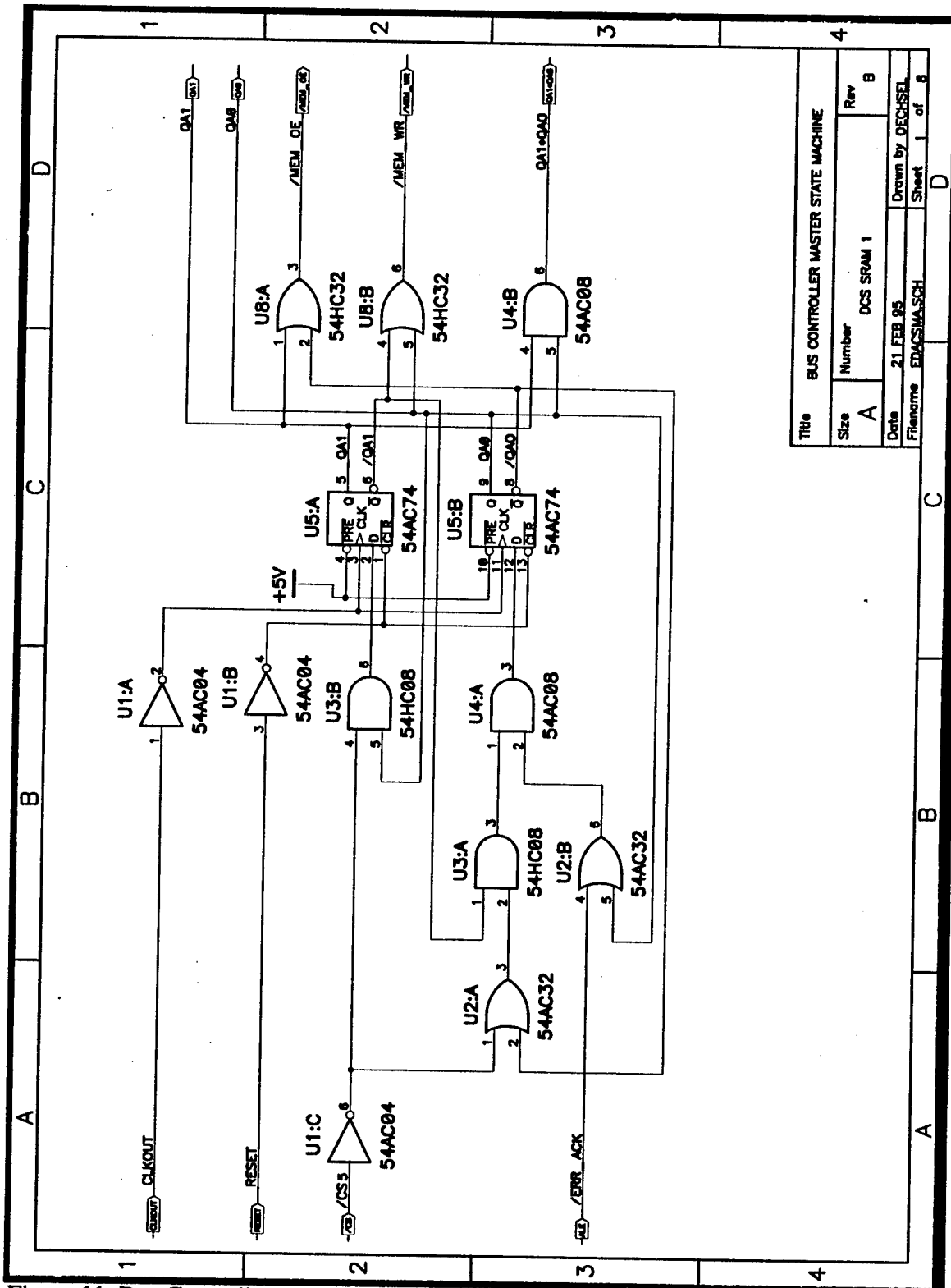
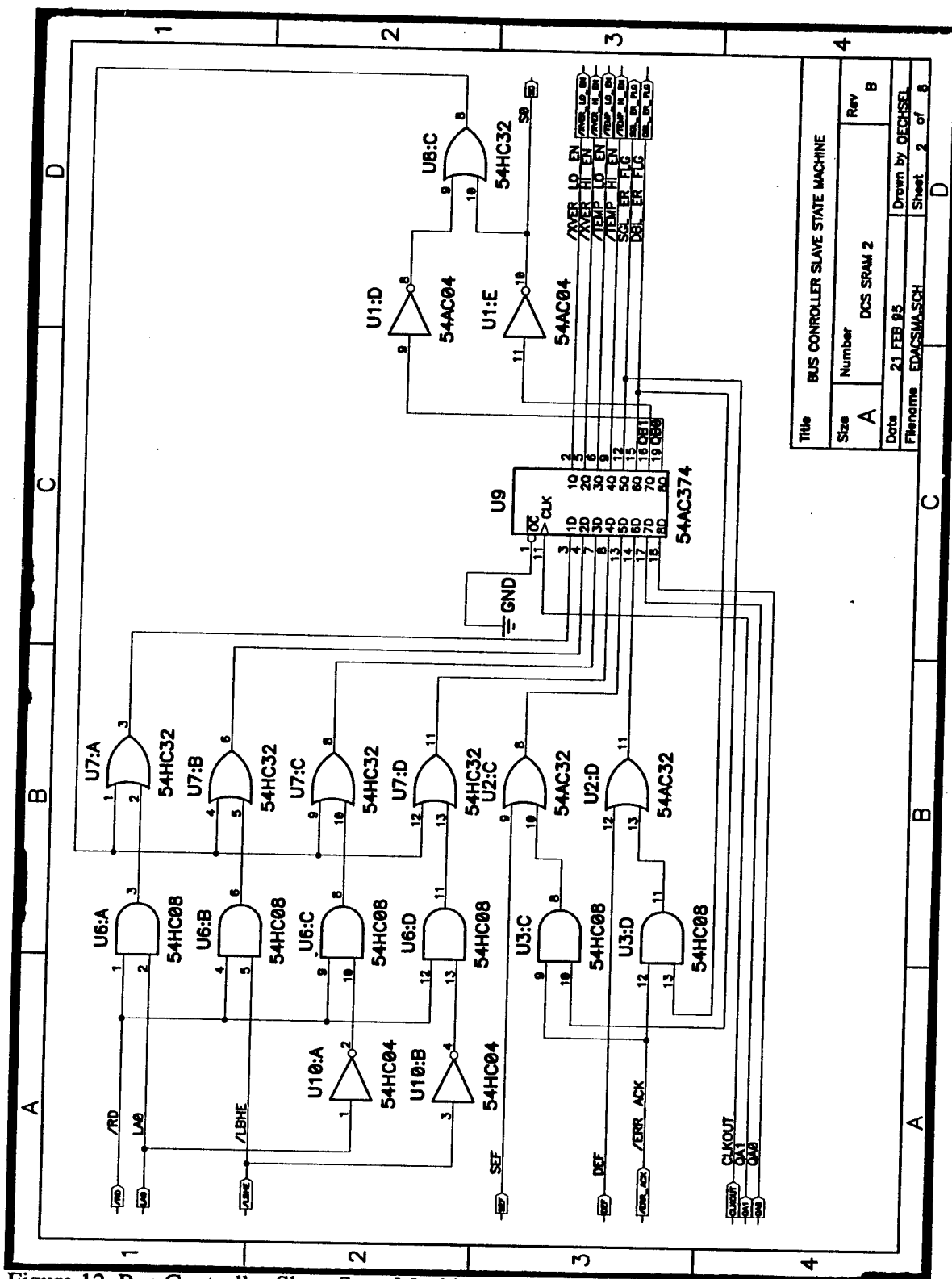


Figure 11 Bus Controller Master State Machine



## 1. Actions Performed in Each State

This machine requires a minimum of six states to provide the control signals necessary to detect and correct an error in memory in one bus cycle. One state must be dedicated to perform each of the following actions sequentially:

1. Address memory and enable memory outputs.
2. Latch the data and check words into the EDAC device and disable the memory outputs.
3. Correct the data word.
4. Move the corrected data word into the temporary register.
5. Enable the proper combination of transceiver and temporary register outputs for the memory access type (see Table 7) and generate a new check word.
6. Latch the data and check words into memory. During a read cycle corrected data is passed to the microprocessor.

A simple option for the state machine is to assign one state to every T-state phase in the bus cycle. In order for this to work, state 5, listed above, must be assigned to T-3-2 so that corrected data is on the data bus ready to be latched into the microprocessor on the T-4-1 clock edge.

A 7.3728 MHz CPU clock provides a CLKOUT signal with a 135.6 ns period as shown in Equation (1).

$$(1) \quad \frac{1}{7.3728 \text{ MHz}} = 135.6 \text{ ns.}$$

Because the rise time of CLKOUT may differ from its fall time, the CLKOUT high time may be different from the CLKOUT low time. The minimum CLKOUT low or high (T-state phase) time is 61.8 ns (see Figure 3, Table 4 and Equation (2)).

$$(2) \quad 0.5 \times 135.6 \text{ ns} - 6 \text{ ns} = 61.8 \text{ ns.}$$

Therefore, the maximum T-state phase time is 73.8 ns as shown in Equation (3).

$$(3) \quad 135.6 - 61.8 = 73.8 \text{ ns}$$

A valid address is available from the M80C186XL 44 ns into T-1-1. The address must then propagate through the address latch to the SRAM address inputs. Data is available from the SRAM 25 ns after the address is valid. Consequently, data is not available until some time into T-1-2. Allowing for 38 ns propagation delay through an HC573 (the address latch), valid data is available to the EDAC circuit no later than 45.2 ns into T-1-2 as shown in Equation (4).

$$(4) \quad 44 \text{ ns} + 25 \text{ ns} + 38 \text{ ns} - 61.8 \text{ ns} = 45.2 \text{ ns}$$

This allows ample time to meet the EDAC circuit's 10 ns setup requirement before the T-2-1 clock edge. Therefore, T-2-1 is assigned to the state in which data is latched in the EDAC and the SRAM outputs are disabled.

The SRAM requires assertion of its output enable signal 25 ns before data is available. This may be accomplished in either T-1-1 or T-1-2. The T-1-1 phase was selected because the logic required to generate the control signal in T-1-1 is simpler (see Control Signals, Section C.6.a, this chapter). By default, the actions performed in each of the remaining T-state phases must be as shown in Table 6.

## **2. State Machine Clock Rate**

Some operations must be completed before continuing to the next state. The maximum clock rate of the bus controller is determined by identifying the longest such operation. If the maximum time required in one state exceeds the minimum time provided by a T-state phase (61.8 ns), faster components would have to be incorporated in the design. Table 13 lists the actions performed in each T-state and shows one T-state phase provides sufficient time.

T-State Phase	Action	Time Req. (ns)	Total Time Req. Before Next State Transition (ns)
T-1-1	Enable SRAM	25 <sup>1</sup>	NA <sup>1</sup>
	Allow for EDAC setup	10 <sup>1</sup>	
T-1-2	No operation	0	0
T-2-1	Latch data into EDAC circuit. Generate EDAC error flags	15.5*	26.1 <sup>2</sup>
	Disable SRAM	10*	
	Error Flags propagate through reset logic	8.6	
	Allow for Error Flag Latch setup	2	
T-2-2	Correct Data	20.5	22.5
	Allow for Temporary Register setup	2	
T-3-1	Latch Data into Temp. Register (hold time)	2.6*	21.5
	Set EDAC to Generate	21.5*	
T-3-2	Enable Transceivers	12.7* <sup>3</sup>	52.2 <sup>3</sup> 37.2 <sup>2,4</sup>
	Enable Temporary Register Outputs	9.5* <sup>3,4</sup>	
	Propagate through Transceiver	7.7 <sup>4</sup>	
	Generate Check Word	24.5 <sup>3</sup>	
	Allow for SRAM setup	15 <sup>3</sup>	
	Allow for M80C186XL data setup	20 <sup>4</sup>	
T-4-1	Latch Data into SRAM (write recovery time)	3	3
T-4-2	Set EDAC to Read	21.5*	21.5
	Disable Transceivers	12.7*	
	Disable Temporary Register Outputs	10.5*	
Notes: *Indicates actions performed in parallel. <sup>1</sup> Not required to be complete until T-2-1 <sup>2</sup> Must be complete prior to next CLKOUT transition <sup>3</sup> Required for data transfer to SRAM from temporary register <sup>4</sup> Required for data transfer to M80C186XL from temporary register			

Table 13 Time Required in Each T-state Phase

### 3. Synchronizing the State Machine to the Bus Cycle

Ideally, the SRAM chip-select signal would be used to synchronize the state machine to the bus cycle. Then the state machine would only exit the standby state during SRAM bus cycles. Unfortunately the chip selects from the M80C186XL are not available until 45 ns into T-1-1. Furthermore, because of the large memory space encompassed by the SRAM devices and the limits imposed by the M80C186 CSU, a single chip-select signal cannot be used to select the SRAM. Instead, multiple chip-select signals must be combined with external logic to generate the SRAM chip-select signal. The delays associated with this logic and the required next state logic make the SRAM chip-select unsuitable for synchronizing the state machine to the bus cycle. Instead, the ALE signal is used to synchronize the state machine controller to the bus cycle. A state transition decision based on the SRAM chip-select signal is made later in the bus cycle to prevent SRAM access during non-SRAM bus cycles.

### 4. State Machine Clock

Eight clock edges are needed to clock the bus controller through its eight states. The microprocessor CLKIN signal provides eight clock edges per bus cycle. However, given that the ALE signal is valid 30 ns into the T-state phase and allowing for two levels of AC next state logic propagation delay, at 9 ns per level, and 5 ns of flip-flop setup, requires that the state flip-flop be clocked no sooner than 53 ns after the beginning of the T-state phase as demonstrated in Equation (5).

$$(5) \quad 30 \text{ ns} + (2 \times 9 \text{ ns}) + 5 \text{ ns} = 53 \text{ ns}$$

This limits the maximum acceptable CLKIN to CLKOUT skew to 15.8 ns, shown in Equation (6).

$$(6) \quad 68.8 \text{ ns} - 53 \text{ ns} = 15.8 \text{ ns}$$

The maximum acceptable skew (15.8 ns) exceeds the maximum M80C186XL specified limit of 25 ns ( $T_{CICO}$  shown in Figure 3 and Table 4). To eliminate this problem the design uses both edges of the CLKOUT signal to provide the required eight state machine clock edges per bus cycle.

Using both edges of the CLKOUT signal requires using two state machines operating in parallel: one clocked by rising edge of CLKOUT, and the other clocked by the falling edge of CLKOUT. Because each state machine is clocked only four times per bus cycle, each machine needs to provide only four states. Furthermore, only one state machine (the master) requires next state logic; the other machine (the slave) simply echoes the state of the master machine at the next clock edge. Because the state machine synchronizes to T-1-1, which corresponds to a falling edge of CLKOUT, the master state machine must be clocked by the falling edge of CLKOUT. Using the CLKOUT signal in this manner adds two gate loads (approximately 20 pF) to the CLKOUT signal; one AC04 (Figure 11, U1:A) and one AC374 (Figure 12, U9).

## 5. Next State Logic

### a. State Assignment

The two state bits used in the master state machine are designated QA1 and QA0. The two state bits making up the slave state machine are designated QB1 and QB0. The states are assigned so that under normal operation only one state bit changes during each state transition. This eliminates the possibility of static hazards under normal operation when using the state bits to generate control signals. The bit sequence used is 00, 01, 11, 10. These states correspond to T-I (or T-4), T-1, T-2, and T-3 respectively. Table 14 shows the state of both state machines in each phase of an SRAM access bus cycle.

T-State Phase	QA1 QA0	QB1 QB0
T-4-2, T-I-1, T-I-2	0 0	0 0
T-1-1	0 1	0 0
T-1-2	0 1	0 1
T-2-1	1 1	0 1
T-2-2	1 1	1 1
T-3-1	1 0	1 1
T-3-2	1 0	1 0
T-4-1	0 0	1 0

Table 14 State Assignment by T-state Phase

***b . Master State Machine State Transition Table, Karnaugh Maps and State Diagram***

The master state machine transition table (Table 15) does not account for the unused combinations of ALE and /CS. The Karnaugh maps (Figure 13) show these “don’t care” inputs based upon the ALE and the chip-select (/CS) waveforms. Under normal conditions ALE will go low during T-1 such that ALE = 1 is a “don’t care” input for the remainder of the bus cycle. Once low, the /CS signal remains low until the end of the bus cycle. Thus, /CS = 1 is a “don’t care” input in T-2 and T-3. These “don’t cares” do not affect the Q0 next state logic. However, the “don’t cares” do simplify the Q1 next state logic by eliminating one Sum-of-Products (SOP) term. This eliminates the need for two logic gates.

ALE	/CS	Current State		Next State	
		QA1	QA0	QA1	QA0
0	X	0	0	0	0
1	X	0	0	0	1
X	0	0	1	1	1
X	1	0	1	0	0
X	X	1	1	1	0
X	X	1	0	0	0

Table 15 Master State Machine Transition Table



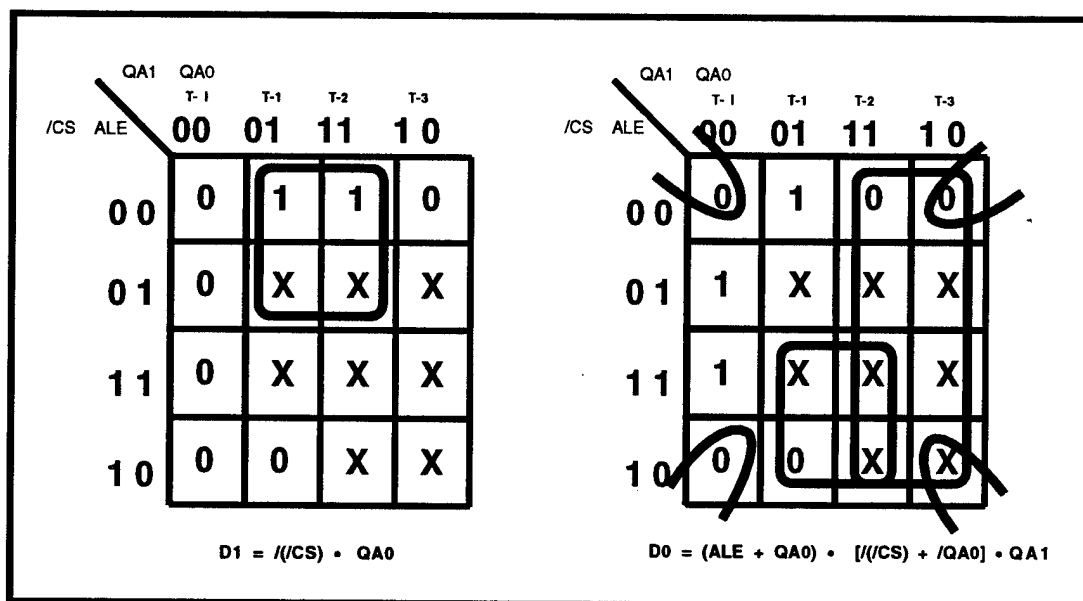


Figure 13 Master State Machine Next State Karnaugh Maps and Equations

The benefit of using fewer gates outweighs the associated risks. The state diagram for the master state machine (Figure 14) shows the transition path for the “don’t care” input in gray scale. This path will only be used when /CS goes high before T-3. The /CS signal could go high as a result of an SEU or a microprocessor glitch. However, the SSI devices used in the next state logic are not susceptible to SEUs and a microprocessor glitch would most likely cause more serious problems such that a missed bus cycle would not matter. Figure 15 is the state diagram for both state machines operating in parallel. The states are shown in QA1 QA0 QB1 QB0 format.

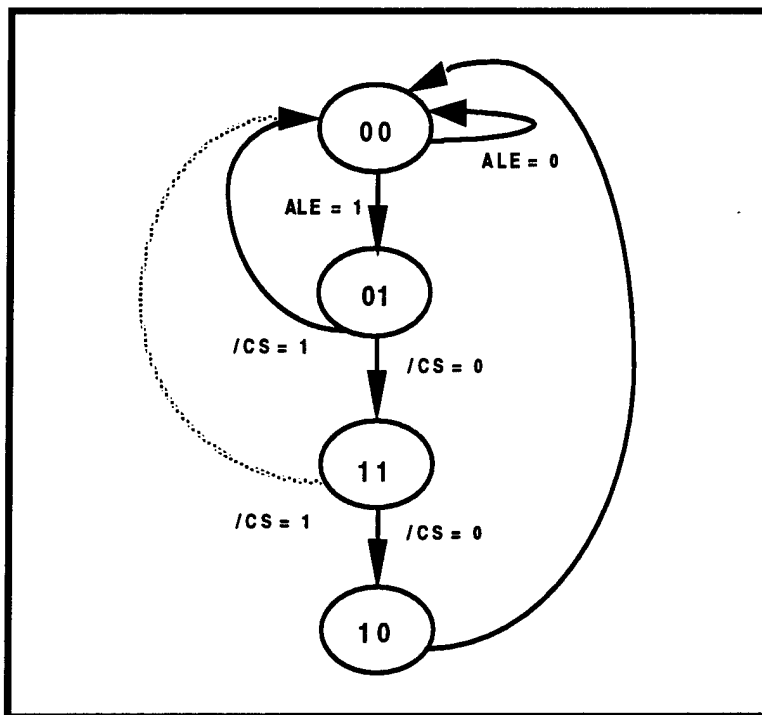


Figure 14 Master State Machine State Diagram QA1 QA0

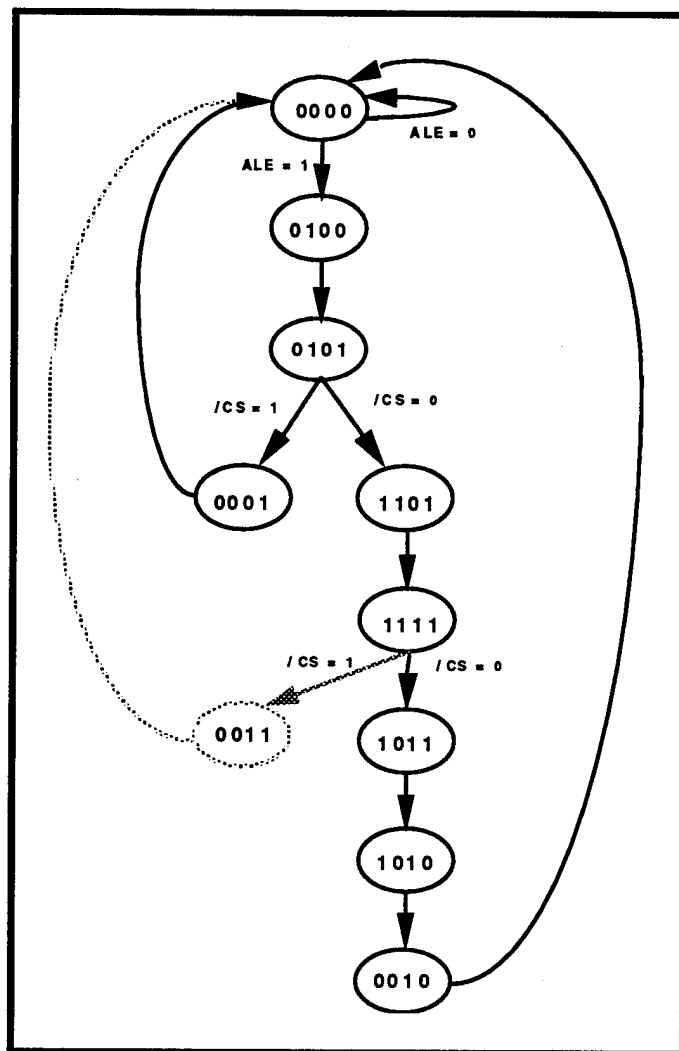


Figure 15 State Diagram QA1 QA0 QB1 QB0

### *c. Timing Verification and Chip-select Considerations*

Figure 11 shows the master state machine. The next state logic includes the gates labeled U1:C, U2:A-B, U3:B and U4:A. The figure shows that the longest feedback path requires three levels of logic. The 135.6 ns period between falling clock edges provides a 53.2 ns margin when allowing up to 23 ns (assuming worst case HC logic) of propagation delay through each logic level, 4.3 ns for flip-flop setup, and 9.1 ns for flip-flop propagation delay as indicated by Equation (7).

$$(7) \quad 135.6 \text{ ns} - [(3 \times 23 \text{ ns}) + 4.3 \text{ ns} + 9.1 \text{ ns}] = 53.2 \text{ ns}$$

The shortest feedback path from one master state machine flip-flop output to the input of the other master state flip-flop includes one level of next state logic. The minimum logic gate propagation delay is 2.2 ns and the minimum flip-flop propagation delay is 2.6 ns. Therefore, the maximum acceptable clock skew between the two master state machine flip-flops is 4.8 ns. Assuming the CLKOUT signal travels at  $\frac{2}{3}$  the speed of light in the PCB trace and ignoring the difference in trigger thresholds between the two flip-flops, Equation (8) shows that there may be a maximum of 0.96 m difference in the length of the CLKOUT PCB traces to each of the master state machine flip-flops.

$$(8) \quad \frac{2}{3} (3 \times 10^8 \text{ m/s}) \times (4.8 \times 10^{-9} \text{ s}) = 0.96 \text{ m}$$

Figure 14 shows that ALE must propagate through only two levels of AC logic. As discussed previously this is acceptable (see State Machine Clock, Section C.4, this chapter). The SRAM chip-select must be available in time to propagate through the next state logic and meet the flip-flop setup time before the T-2-1 clock edge. Figure 14 shows that the longest chip-select signal path includes an AC04 (5.9 ns), an HC08 gate (23 ns), and an AC08 gate (7.9 ns). The AC74 flip-flop requires 4.3 ns for setup and the M80C186XL chip-selects are valid 45 ns into T-1-1. Thus, there are 41.1 ns available for SRAM external chip-select logic. This is shown in Equation (9).

$$(9) \quad 135.6 \text{ ns} - (45 \text{ ns} + 5.7 \text{ ns} + 8.6 \text{ ns} + 23 \text{ ns} + 7.9 \text{ ns} + 4.3) = 41.1 \text{ ns}$$

If more time is required, the HC08 may be replaced by an AC08, contributing 15.1 ns.

## 6. Control Signal Logic

The state machine provides control signals for the SRAM, EDAC circuit, temporary register, and bus transceivers. Additionally, the two EDAC circuit error flags are latched to provide interrupt request signals to the microprocessor to flag SRAM errors.

### *a. General Information*

Some of the control signals offer a degree of flexibility about how long they are asserted. For instance, the temporary register latches data when the /TEMP\_LE signal goes low. The /TEMP\_LE signal must be high during the state prior to when data is latched so it can be brought low appropriately. Exactly when /TEMP\_LE is brought high does not affect system operation. If a control signal can be asserted for an even number of CLKOUT transitions, that signal can be derived from the state bits of just one of the state machines. All the control signals in this design are asserted for an even number of CLKOUT transitions.

### *b. General Timing Considerations*

There are two critical timing parameters used in this design: the minimum Time Before State Change ( $T_{BSC}$ ) and the minimum Time Before Next Clock ( $T_{BNC}$ ). There is a subtle difference between the two parameters.  $T_{BSC}$  is measured from when the state bits become stable in one state to when the state bits are stable in the following state.  $T_{BNC}$  is measured from when the state bits are stable to the next CLKOUT edge. These parameters are different for the two state machines for the following reasons:

- The CLKOUT signal is not perfectly symmetric.
- The flip-flops used to store the master state bits are a different type than the flip-flops used to store the slave bits.
- The master state machine is clocked off an inverted CLKOUT (/CLKOUT). The inverter makes the CLKOUT signal appear even less symmetric to the two state machines.

A conservative method was used to calculate these two parameters. When calculating the  $T_{BSC}$ , the maximum propagation delay of the current state flip-flop was used with the minimum propagation delay of the next state flip-flop. This situation is improbable because

both flip-flops will be operating in the same environment, but the method used allows the designer to feel more comfortable when some timing margins are only a few nanoseconds.

Accounting for the CLKOUT propagation delay through an AC04 (min 1.7 ns, max 5.9 ns) and the AC74 flip-flop propagation delay (min 2.6 ns, max 9.1 ns), the master state machine state bits, QA1 and QA0, will transition 4.3 ns to 15 ns after the falling edge of CLKOUT. The slave state machine state bits, QB1 and QB0, will transition 2.9 ns to 10.2 ns (AC374 flip-flop delay) after the rising edge of CLKOUT. Given at least 61.8 ns between clock edges, the minimum  $T_{BNC}$  for the master state machine ( $MSMT_{BNC}$ ) is 46.8 ns as shown in Equation (10).

$$(10) \quad 61.8 \text{ ns} - 15 \text{ ns} = 46.8 \text{ ns}$$

The minimum  $T_{BSC}$  for the master state machine ( $MSMT_{BSC}$ ) is 49.7 ns, shown in Equation (11).

$$(11) \quad 61.8 \text{ ns} - 15 \text{ ns} + 2.9 = 49.7 \text{ ns}$$

Similarly, the minimum  $T_{BNC}$  for the slave state machine ( $SSMT_{BNC}$ ) is 52 ns, shown in Equation (12).

$$(12) \quad 61.8 \text{ ns} - 9.8 \text{ ns} = 52 \text{ ns}$$

The minimum  $T_{BSC}$  for the slave state machine ( $SSMT_{BSC}$ ) is 56.3 ns, shown in Equation (13).

$$(13) \quad 61.8 \text{ ns} - 9.8 \text{ ns} + 4.3 \text{ ns} = 56.3 \text{ ns}$$

### ***c. Memory Output Enable***

The memory output must be enabled in T-1-2 so that data from memory may be latched into the EDAC circuit in T-2-1. Additionally the memory outputs must be disabled in T-2-1 so that the EDAC circuit can drive the data bus in T-2-2. Enabling the memory outputs in T-1 does not affect system operation. The /MEM\_OE signal will be asserted whenever the ALE signal goes high forcing the master state machine to the 01 state despite whether the upcoming bus cycle is an SRAM access. However, if the upcoming bus cycle is not an SRAM access, the SRAM chip-select will not be asserted. The SRAM

chip-select has priority over the output enable ensuring that the SRAM outputs are only enabled during SRAM bus cycles. The  $\overline{\text{MEM\_OE}}$  signal is derived as shown in Equations (14) and (15).

$$(14) \quad \overline{\text{MEM\_OE}} = 0 \text{ if } \overline{\text{QA1}} \cdot \text{QA0} = 1; \text{ else } \overline{\text{MEM\_OE}} = 1$$

$$(15) \quad \therefore \overline{\text{MEM\_OE}} = \overline{\overline{\text{QA1}} \cdot \text{QA0}} \\ = \text{QA1} + \overline{\text{QA0}}$$

As previously discussed,  $\overline{\text{MEM\_OE}}$  is asserted one state earlier than required providing ample time for EDAC circuit setup (see Table 13). Subtracting 23 ns for propagation delay through an HC32 gate and 10 ns to disable the SRAM from 49.7 ns ( $\text{MSMT}_{\text{BSC}}$ ) results in a 16.7 ns margin.

#### *d. Memory Write*

The memory write control signal ( $\overline{\text{MEM\_WR}}$ ) must present a rising edge in T-4-1 to latch the data into memory. The  $\overline{\text{MEM\_WR}}$  signal may be brought low any time after T-2-1, when data from memory is latched into the EDAC. To assert the  $\overline{\text{MEM\_WR}}$  for an even number of CLKOUT transitions it is asserted in T-3-1. The  $\overline{\text{MEM\_WR}}$  signal is derived as shown in Equations (16) and (17).

$$(16) \quad \overline{\text{MEM\_WR}} = 0 \text{ if } \text{QA1} \cdot \overline{\text{QA0}} = 1; \text{ else } \overline{\text{MEM\_WR}} = 1$$

$$(17) \quad \therefore \overline{\text{MEM\_WR}} = \overline{\text{QA1} \cdot \overline{\text{QA0}}} \\ = \overline{\text{QA1}} + \text{QA0}$$

Subtracting 23 ns for propagation delay through an HC32 gate and 3 ns for memory write recovery time from 49.7 ns ( $\text{MSMT}_{\text{BSC}}$ ) results in a 23.7 ns margin.

#### *e. EDAC Control*

The EDAC circuit requires two control inputs: S0 and S1. Table 16 shows the required state of both control signals in each T-state phase. Comparing Table 16 to Table 14 yields the required Boolean expressions for S0 and S1, shown in Equations (18) and (19).

T-State Phase	EDAC Function	S 1	S 0	Error Flags
T-4-2, T-I-1, T-I-2	READ	0	1	0
T-1-1	READ	0	1	0
T-1-2	READ	0	1	0
T-2-1	LATCH	1	1	VALID
T-2-2	CORRECT	1	0	VALID
T-3-1	GENERATE	0	0	0
T-3-1	GENERATE	0	0	0
T-4-1	GENERATE	0	0	0

Table 16 EDAC Control Signals and Error Flags by T-state

$$(18) \quad S0 = \overline{QB1}$$

$$(19) \quad S1 = QA1 \cdot QAO$$

Allowing 21 ns for propagation delay through an HC inverter and 22.5 ns for the longest action initiated by S0 in the EDAC circuit, the action will complete 12.8 ns before the next state transition as shown in Equation (20).

$$(20) \quad 56.3 \text{ ns} - 21 \text{ ns} - 22.5 \text{ ns} = 12.8 \text{ ns}$$

An AC08 gate (7.9 ns) must be used to generate the S1 signal because the error flags that are valid 15.5 ns after the S1 signal goes high must pass through one AC32 gate (8.6 ns) and meet the setup time for the AC374 flip-flop (2 ns) within 46.8 ns ( $MSMT_{BNC}$ ). This allows a margin of 12.8 ns, as shown in Equation (21).

$$(21) \quad 46.8 \text{ ns} - (7.9 \text{ ns} + 15.5 \text{ ns} + 8.6 \text{ ns} + 2 \text{ ns}) = 12.8 \text{ ns}$$

This margin is not enough to absorb the extra propagation delay incurred when using HC gates.



### f. Memory Error Flags

The EDAC circuit error flags (SEF and DEF) are valid only during T-2. They must be latched to provide interrupt request signals to the microprocessor that remain asserted until acknowledged by the microprocessor. Figure 16 shows the Karnaugh maps and resulting Boolean expressions to control a D-type latch to provide the required interrupt request signals. The latched single error flag (SGL\_ER\_FLG) and latched dual error flag (DBL\_ER\_FLG) are set (high) on the clock edge after the respective EDAC circuit error flag goes high. They remain asserted until the microprocessor acknowledges them by asserting (low) the error acknowledge strobe (/ERR\_ACK). If an /ERR\_ACK arrives at the latch coincidentally with a new EDAC circuit error flag, the latch remains set to indicate another memory bit-error. The resultant logic circuits are shown in Figure 12 gates U2:C, U2:D, U3:C, and U3:D.

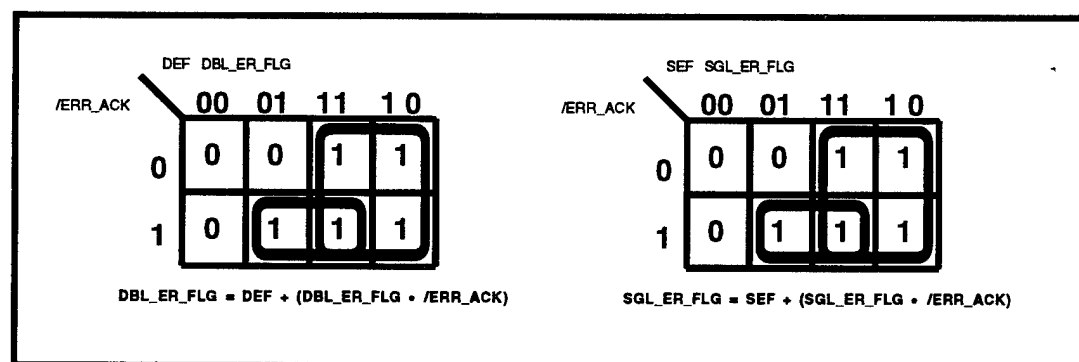


Figure 16 Error Flag Latch Control

The AC374 octal flip-flop (U9 in Figure 12) used to store the slave state bits has two flip-flops available to latch the error flags. However, because the EDAC error flags are valid for only one T-state, using these flip-flops requires latching the EDAC circuit error flags on the CLKOUT edge after they become valid. This allows little time for the EDAC error flags to propagate through the reset logic. Equation (21) shows a 12.8 ns margin when AC logic is used in the EDAC error flags' data paths. This margin is not large enough to absorb the increased propagation delay incurred when using HC logic.

#### ***g. Temporary Latch Enable***

The temporary register latch enable control signal (/TEMP\_LE) must be high in T-2-2 so it can go low in T-3-1 to latch data into the temporary register. The /TEMP\_LE signal must remain low through T-4-1 to hold the data in the temporary register. From T-4-2 through T-2-1 /TEMP\_LE is a “don’t care” signal. The state bit QA0 alone would provide an adequate control signal. However, when the master state machine toggles between the 00 state and the 01 state during bus cycles that are not SRAM accesses, the /TEMP\_LE would also toggle. Using  $\text{/TEMP\_LE} = S1 = QA1 \cdot QA0$  eliminates the extra toggle without adding any gates. Allowing 7.9 ns for propagation delay through the AC08, data will be latched into the temporary register well before the next state transition.

#### ***h. Data Bus Transceiver / Temporary Register Output Enables***

The earliest that the data bus transceivers or temporary register may be enabled without causing data bus contention is T-3-2. As shown in Table 14 the output enable control signals must be asserted 37.2 ns before the next CLKOUT edge and 52.2 ns before the /MEM\_WR signal goes high at the next state transition. With only 56.3 ns  $T_{BSC}$  before the next state transition, there is not enough time to generate the appropriate output enable signals during T-3-2. Instead, they must be generated early and released at the T-3-2 clock edge. This provides a margin of 4.1 ns as shown in Equations (22) and (23).

$$(22) \quad 56.3 \text{ ns} - 37.2 \text{ ns} = 19.1 \text{ ns}$$

$$(23) \quad 56.3 \text{ ns} - 52.2 \text{ ns} = 4.1 \text{ ns}$$

Table 7 shows the required output enable signals by memory access type. The control signals Latched A0 (LA0), Latched Bus High Enable (/LBHE), and either READ (/RD) or WRITE (/WR) from the microprocessor determine the memory access type. Using the READ signal instead of the WRITE signal requires one less inverter but adds a four-gate load to the READ signal. Using the WRITE signal adds only one gate load to the WRITE signal. The M80C186XL timing specifications are given for a maximum capacitive load ( $C_L$ ) of 200 pF. A CMOS gate load is typically 10 pF. To keep the part count at a minimum, this design uses the READ signal. The four-gate load still allows a 160 pF margin for other peripherals.

Figure 17 shows the Karnaugh maps and corresponding Boolean expressions for the output enable control signals. The resulting logic circuit is shown in Figure 15: gates U6:A-D, U7:A-D, and U10:A-B. The output enable signals must be qualified by a signal allowing them to go low and reset the flip-flops only at the T-3-2 clock edge. This is accomplished by OR-ing the output enable signals with a signal that only goes low T-2-2 through T-3-2. This signal is  $\overline{QB1} + \overline{QB0}$ . The qualifying circuit is shown in Figure 15: gates U1:D-E, U8:C and U7:A-D.

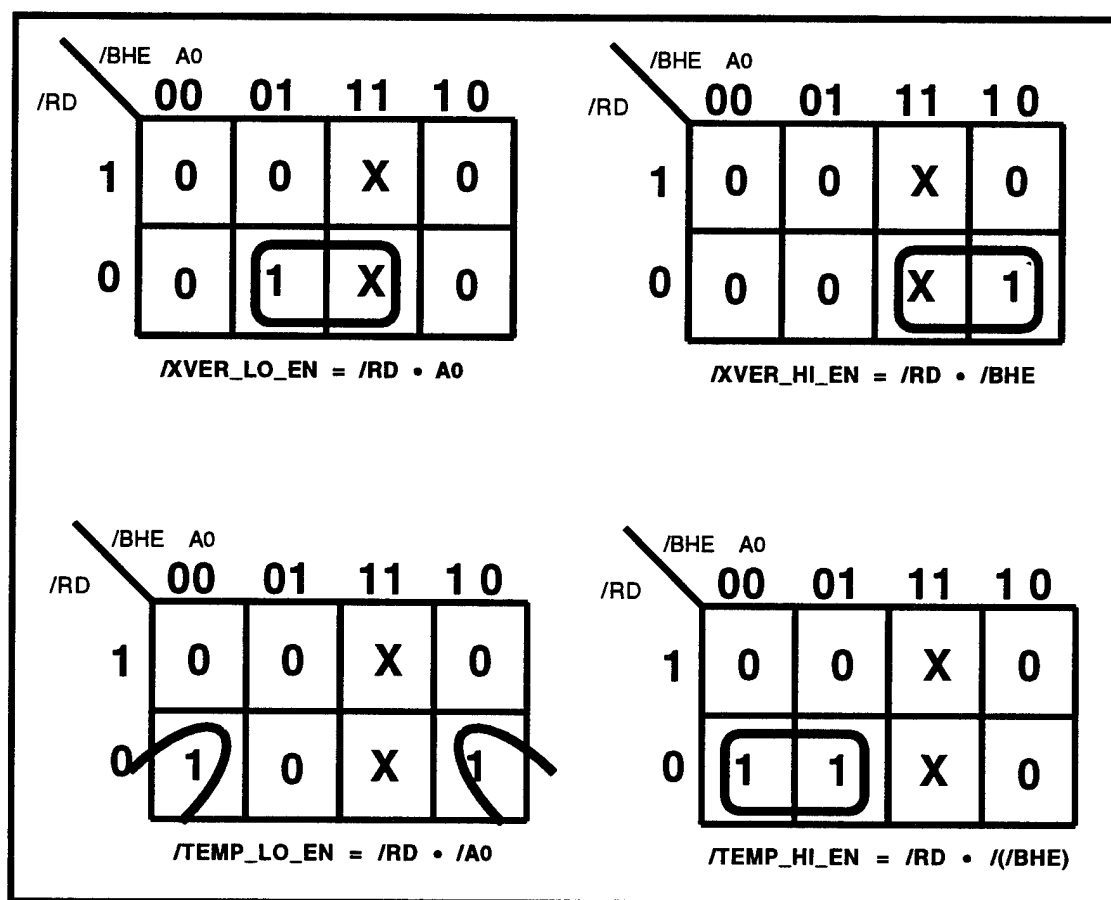


Figure 17 Bus Transceiver and Temporary Register Output Enable Control Signals

The feedback path contains three levels of logic requiring at most 23 ns of propagation delay per level. An additional 10.2 ns and 2 ns are required for flip-flop propagation delay and setup time providing a 54.4 ns margin, as shown in Equation (24).

$$(24) \quad 135.6 \text{ ns} - [(3 \times 23 \text{ ns}) + 10.2 \text{ ns} + 2 \text{ ns}] = 54.4 \text{ ns}$$

The LA0 and /LBHE are available in T-1-2 and the /RD signal is available during T-2-1 allowing plenty of time for propagation through the three (at most) levels of HC logic and flip-flop setup before the T-3-2 clock edge.

#### *i. Bus Transceiver Direction Control*

The bus transceiver directional control must be valid for the bus cycle type (read or write) during T-3-2 through T-4-2 while the transceiver outputs are enabled. Figures 4 and 5 show that M80C186 Direction Control signal (DT-/R) is valid for each type of bus cycle T-1-1 through T-4-2 and may control the data bus transceivers directly.

### **7. Other Design Considerations**

#### *a. Bus Hold Time*

During an SRAM bus cycle there are two occasions when data on the bus is latched at the same time that the source of the data is disabled:

- During T-2-1 data from memory is latched into the EDAC circuit and the memory outputs are disabled.
- During T-3-1 corrected data from the EDAC circuit is latched into the temporary register and the EDAC is set to generate.

This design relies on the time constant of the data bus to provide the required hold times. The longest data hold time required by either the EDAC circuit or the temporary register (AC573) is 5 ns. The bus hold time was measured and found to be approximately 20  $\mu$ s. (See Chapter IV, Section E.)

#### *b. System Power Up*

Upon system power up,  $V_{CC}$  and CLKIN will reach the microprocessor at approximately the same time.  $V_{CC}$  will also reach the memory system at approximately the

same time as the microprocessor, but CLKOUT will reach the memory system some time later. The system will self synchronize within  $3\frac{1}{2}$  CLKOUT cycles and reach the standby state (0000) within five CLKOUT cycles. During the time between power up and synchronization the control signals are in a random state and more than one bus driver may be enabled resulting in bus contention. An asynchronous reset signal is available to force the master state machine to the 00 state, but still up to three CLKOUT cycles are required to synchronize the slave state machine and its associated outputs. Three methods were considered to deal with this problem.

The first method considered was using an asynchronous signal to tri-state the slave state machine and its associated outputs during system start-up. Then pull-up and pull-down resistors could be used to force control signals to the desired states. The problem with this method is that given a 10 pF line capacitance, the 1 k $\Omega$  resistors required to keep the time constant at 100 ns, would increase power requirements during normal operation. Larger resistors would consume less power but increase the time constant such that the system could self-synchronize before the tri-stated outputs are pulled up or down as required.

The second method considered was to use an asynchronous logic signal asserted during start-up to force the output enable signals to their unasserted state. However this method still requires up to two CLKOUT signals to ensure the output enable signals are not asserted. Furthermore this method requires at least four more gates.

The third method considered was to simply let the system self synchronize. The Harris Product Selection Guide states that no danger of latchup results from backdriving (short circuiting) AC CMOS outputs. However, to prevent overheating the IC, Harris recommends that only one output be backdriven at a time for no more than one second. [Ref. 10: p 3-11] Assuming a linear relationship, back driving one output for one second would dissipate 125,000 times the heat as backdriving 16 outputs (as on ACS630MS) for 500 ns. Considering the few times the system is expected to be powered up during its lifetime and the simplicity of this method against the limited improvement offered by the other methods, this final method was used in this design.

The schematic shown in Figure 11 and Appendix C shows the RESET signal from the microprocessor connected to the master state machine flip-flop reset through inverter U1:B. However, this arrangement does not offer any advantage over simply tying the reset signal high (unasserted). By the time the M80C186 has asserted the reset signal (up to 28 CLKIN cycles after CLKIN and V<sub>CC</sub> stabilize) during power up, the

flip-flop has already reached the 00 (standby) state. Furthermore, during a warm reset the system will already be synchronized and will automatically reach the standby state within four CLKOUT cycles without inducing bus contention.

## **V. MEMORY SYSTEM SOFTWARE REQUIREMENTS**

### **A. SRAM INITIALIZATION**

Interrupt requests are generated based on the errors detected during the “read and correct” operation performed during every SRAM access. If a write is performed to an “empty” SRAM address, i.e., an address that has not had data stored in it since power up, the random data currently at that address is checked for errors before being overwritten. The random check word read is usually not the proper check word for the random data word read and the EDAC circuit generates false error flags. To avoid this situation, the SRAM must be initialized as part of the microprocessor start-up. The SRAM initialization process involves writing arbitrary data to every SRAM address to set the check words so that the first time meaningful data is written to an SRAM address false error flags are not generated. Before the memory error interrupts are unmasked, the /ERR\_ACK signal should be asserted to clear any pending memory error interrupt requests. This requires that both memory error interrupt requests be maskable.

The preliminary design of the DCS recommends that the dual error interrupt is assigned to the Non-Maskable Interrupt (NMI) [Ref. 11, p. 36]. However using the NMI for dual errors offers no advantage over using the highest priority maskable interrupt and it forces the requirement to use hardware to prevent dual error interrupt requests during SRAM initialization.

### **B. INTERRUPT TRIGGER LEVEL**

An asynchronous error acknowledged from the M80C186XL clears the interrupt request. If the error acknowledge signal and another EDAC error flag occur during the same CLKOUT cycle, the interrupt request remains asserted. So that this second interrupt request is recognized, the M80C186XL must be programmed so that the memory error interrupts are level triggered.

### **C. INTERRUPT ACKNOWLEDGE / SINGLE-BIT ERROR SERVICE ROUTINE**

The M80C186 clears the interrupt request by asynchronously asserting an error acknowledge signal using an I/O mapped register. The interrupt acknowledge signal must

be asserted (low) through at least one CLKOUT cycle to ensure the memory error flag latches are reset.

All single bit errors in memory are corrected without microprocessor action. Thus, single bit error interrupts are not *required* for proper system operation. However, interrupts provide a means to count the number of single-bit memory errors experienced by PANSAT without additional hardware. A limitation to using interrupts to count the errors is that any additional single-bit errors detected while a single-bit error interrupt request is pending will not be counted. However, considering the expected frequency of errors, the number of errors not counted should not be significant.

#### **D. DUAL-BIT ERROR INTERRUPT SERVICE ROUTINE**

To reduce the chances of system failure, the dual-bit error interrupt service routine should not be stored in the SRAM because the SRAM is susceptible to dual-bit errors. Instead, the dual-bit error interrupt service routine should be stored in the DCS radiation hardened Programmable Read Only Memory (PROM). Ideally, to protect the address of the dual-bit error service routine from dual-bit errors, the interrupt vector for the dual-bit error should also be stored in radiation hardened memory. However, this is not practical and the probability of dual-bit errors in either of the two words of the interrupt vector is low enough to make this risk acceptable. Furthermore, executing code at a bad address will probably result in the microprocessor stalling due to an unrecognized instruction. This is not an unrecoverable condition; an external timer will switch control to the other DCS after a predetermined period of microprocessor inactivity.

#### **E. MEMORY WASH**

To reduce the chances of a dual-error due to the accumulation of single-bit errors, each address in the SRAM should be accessed as often as practicable. A background operation that simply reads small consecutive areas of SRAM every time it is invoked will keep errors from accumulating in memory without interfering with other microprocessor functions.



## **VI. SYSTEM TESTING AND PERFORMANCE**

The memory system described in this thesis was fabricated on a wire wrap development board (see Appendix C). This limited the testing to a simple verification of proper system operation. Thermal testing will be completed when the design is implemented on a PCB. At the current stage of development, all tests suggest that the system is operating properly. This chapter describes the tests performed and their results.

### **A. BACKGROUND INFORMATION**

The Microtek MICE-III In-Circuit Emulator (ICE) was used to emulate the M80C186XL during initial tests. The emulator provided a means to run a program in emulation RAM to test SRAM on the development board. The emulator allowed rapid modification of the test programs without having to re-program the system Electronically Programmable Read Only Memory (EPROM) on the development board after each modification. The emulator provided an interface that allowed single stepping through the test programs. Once a test program was working properly, EPROMs could be programmed and the test repeated replacing the emulator with an M80C186XL.

The emulator exhibited unusual behavior during testing. When using the emulator to run a program that did not contain an interrupt table, the data in certain areas of memory was corrupted. The addresses corrupted would vary between runs of the program but were always in the last 16 bytes of one of the 64K segments of the memory space. When an interrupt table was added to the operating system this phenomenon disappeared. This was attributed to the emulator because the same phenomenon was also observed when evaluating another development system with non-EDACed RAM.

### **B. DATA INTEGRITY TESTS**

The capability to perform read and write operations to every memory location in SRAM while maintaining data integrity was verified by running a simple test program that first wrote a value to every SRAM address then read every SRAM address and compared the value read with the value written. The test was performed by first writing the byte value 0 to all SRAM cells, and then comparing all the written cells. This was repeated for all byte values (0 - 255). A similar test was performed using word values. The number of possible word values prohibits testing every word at every memory address. However,

every memory address was accessed with multiple test words. No mismatches were observed.

### **C. EDAC TEST**

The error detection and correction capability was tested by disconnecting one bit of the data bus from memory and grounding it through a 10 k $\Omega$  resistor. This forced a logical zero on the disconnected bit of the bus whenever the bus was not being driven or when the SRAM was driving the bus. This configuration is the electrical equivalent of a bit stuck at zero at every SRAM address.

The test program consisted of a loop that executed a word write with a one in the appropriate stuck bit position to every available memory address. Not all the addresses were available because an interrupt table was required so that the memory error interrupt requests could be counted. The data words were then read back and compared to the value written. Then the process was repeated with another data value that still had a one in the stuck bit position.

This test resulted in no mismatches suggesting that error correction was working properly. Furthermore, once the memory was initialized with “stuck-bit” during the first loop the number of data single-bit error interrupts in each successive loop equaled the total number of memory accesses in the loop. This shows that every interrupt request was recognized. No dual bit error interrupts were observed.

### **D. MEASUREMENT OF DUAL-BIT ERRORS IN UNINITIALIZED MEMORY**

When first powered up the SRAM is in a completely random state. With six random check bits per check word, the percentage of random check words that are the correct check word for the random data word is 1.5625%, shown in Equation (25).

$$(25) \quad \frac{1}{2^6} = 1.5625 \%$$

The number of dual-bit errors expected in 228864 accesses to uninitialized memory is 225288 as shown in Equation (26).

$$(26) \quad 228864 \times \left(1 - \frac{1}{2^6}\right) = 225288$$

The dual-bit errors detected in 228864 accesses to uninitialized memory during 106 start-up cycles were recorded. The average number of dual-bit errors detected was 225267, less than what was expected by 21 errors (0.01%).

## E. DATA BUS SIGNAL HOLD TIME MEASUREMENT

The minimum voltage guaranteed to be recognized as a logical one by the Harris CMOS components used in this design is 3.5 V when  $V_{CC} = 5.0$  V ( $V_{IH} = 0.7 \times V_{CC}$ ). Figure 18 shows an oscilloscope trace of the decay of a logical one on the bus with all the bus drivers tri-stated. The dashed horizontal line is at 3.5 V and the dashed vertical line is at 20.8  $\mu$ s indicating that the bus hold time is approximately four thousand times more than required to meet the required 5 ns hold time. The time constant of the data bus may change once implemented on a PCB. However, most of the bus capacitance and resistance are caused by the devices on the bus and not the bus itself so that the time constant should not change significantly.

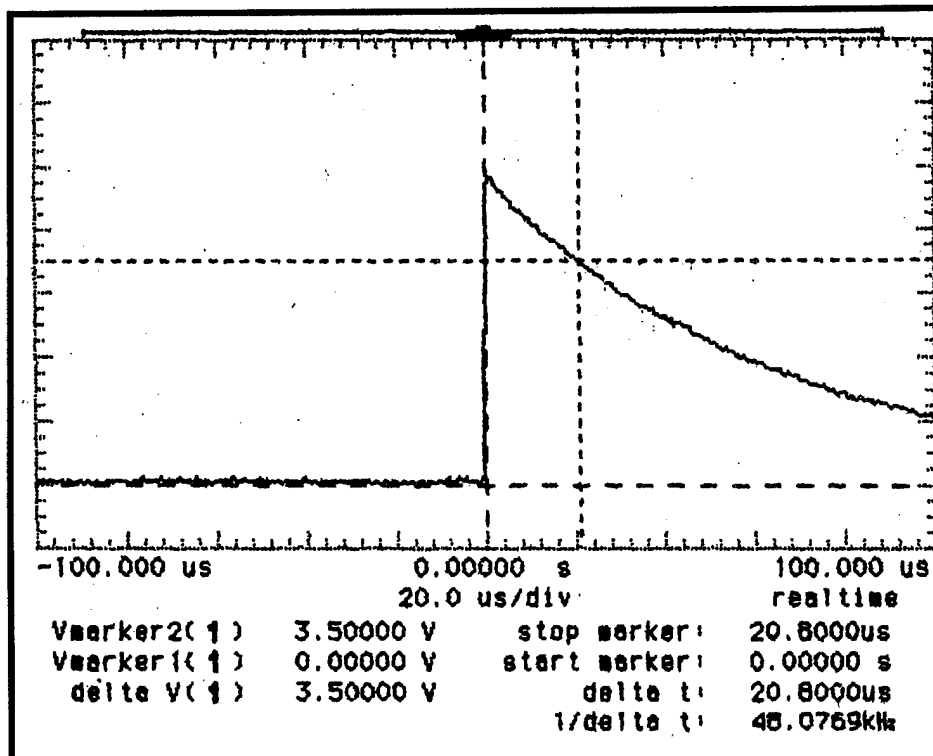


Figure 18 Data Bus Signal Decay



## VII. RECOMMENDATIONS AND CONCLUSION

### A. RECOMMENDATIONS

#### 1. Short Term Recommendations

For reasons discussed in this thesis, the inverter (U1:B) used to reset the master state machine should be eliminated. Instead the flip-flops (U5:A-B) reset input should be connected directly to  $V_{CC}$ .

After the remaining DCS subsystems' designs are completed, if there are any unused CMOS (AC or HC) inverters, AND gates and OR gates in the DCS, the memory write control circuit should be modified to eliminate the write back of data during read cycles when the single-bit error latch is not set. Figure 19 provides the Karnaugh map and Boolean expression to accomplish this. Figure 20 shows a logic circuit that will meet the timing requirements for the  $\overline{\text{MEM\_WR}}$  signal with either family of CMOS logic.

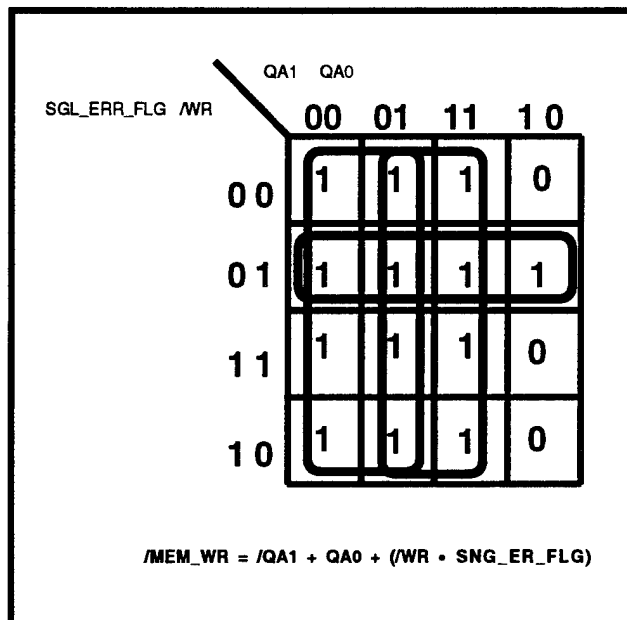


Figure 19 Improved  $\overline{\text{MEM\_WR}}$

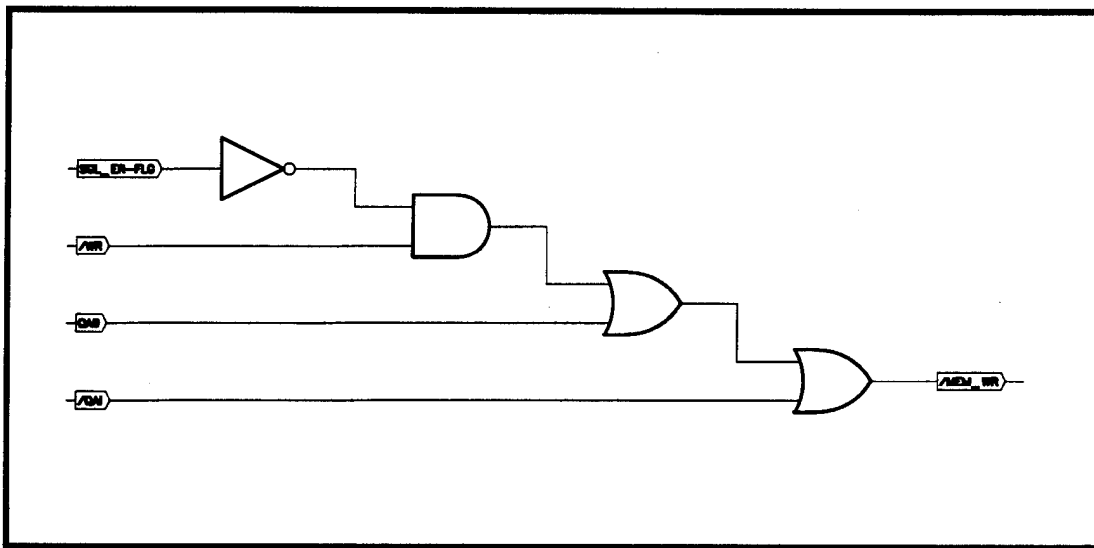


Figure 20 Improved /MEM\_WR Control Logic Circuit

The next step in the memory system design process is to implement this design on a prototype PCB. Before this is done, the M80C186XL chip-select mapping should be completed so that the necessary external SRAM chip-select logic can be included on the PCB. The DCS Programmable Peripheral Interface (PPI) that will provide the /ERR\_ACK signal should also be included on the PCB. The following recommendations are submitted to facilitate the transition from the wire wrap development board to the prototype PCB:

- To improve PCB layout, the Harris CD54AC574 Octal Three State D-type Flip-flop should be used in place of the CD54AC374 Octal Three State D-type Flip-flop. The only difference between the two devices is the pin out. The AC574 was not used for development because of a long lead time required to obtain this part. However, AC574s have been ordered and should arrive in time for the first PCB prototype.

- The /CLKOUT PCB traces to each of the master state machine flip-flops should be kept as close in length as possible to minimize clock skew (see Equation (8)).
- When laying out the DCS PCB, the SRAM data bus and check word bus should be as short as possible to reduce the effects of signal reflections on the unterminated buses. Harris recommends terminating data paths that exceed five inches [Ref. 12, p. 1].

Thermal testing should be performed on a PCB prototype to verify proper operation over the temperature range that PANSAT may encounter.

The SRAM wash, single-bit memory error, and dual-bit error service routines require further investigation. The material in Appendix A should be referred to when considering the SRAM wash duty cycle. The priority level of each interrupt must also be considered.

## **2. Long Term Recommendations**

The design described in this thesis should be used as a starting point in the design of an Application Specific Integrated Circuit (ASIC) for the follow-on to the PANSAT experiment. Either a radiation hardened custom device or a high speed, radiation hardened Programmable Gate Array (PGA) could be used to implement both the EDAC circuit and bus controller in one package. The inherent flexibility and lower propagation delays of an ASIC would eliminate many of the design tradeoffs accepted in this "glue logic" implementation while reducing the PCB area required. Suggested modifications to the current design for an ASIC implementation are:

- Use the microprocessor status bits (S2:0) and the /BHE and A0 signals, all available in T-1-1, to detect the bus cycle type. If word write operations can be identified early in the bus cycle the unnecessary "read and correct" and the associated extraneous error flags may be eliminated.
- Use an EDAC circuit that generates a corrected check word when it corrects the data word. Then the corrected data can be written to memory without the intermediate step of moving the corrected data word to the temporary register so that it can be sent back to the EDAC circuit to generate the new check word. If a syndrome is needed, use another output port rather than the check word I/O port.

## **B. CONCLUSION**

The memory system design described in this thesis provides PANSAT with a robust SRAM system suitable for use in the LEO environment. The system met the design requirements by using a minimum number of readily available, reasonably priced, CMOS components. Design decisions made to achieve a fast, compact design add to the power required by the system. However, the low power consumption of CMOS components makes this an acceptable tradeoff.



## APPENDIX A. ESTIMATION OF UNCORRECTABLE MEMORY ERRORS ABOARD PANSAT

The following estimation is based on the SEU rate observed in the SRAMs aboard UoSAT-2, a satellite placed in a 700 km, polar orbit in 1984. UoSAT-2 used several different types of unhardened RAMs. SEU rates in both static and dynamic RAMs of varying densities from several manufacturers were observed. This satellite employed EDAC with an eight minute RAM wash cycle to track the number of memory errors and the approximate location in the orbit where each error occurred. The average SEU rate observed in the CMOS SRAMs was approximately  $10^{-6}$  SEUs/bit/day and the variation in upset rate on a monthly basis was about one order of magnitude [Ref. 7, pp. 2340-2342]. Most (>75%) of the SEUs observed occurred in the SAA. [Ref. 13, p. 1944]. The CMOS SRAMS exhibited no significant difference in (SEU) sensitivity for different types, architectures and manufacturers. [Ref. 7, p 2340]

For this estimation the following assumptions are made:

- PANSAT will experience  $10^{-5}$  SEUs/bit/day.
- All SEUs will occur in the SAA.
- PANSAT will pass through the SAA four times per day (a conservative estimate)
- SEUs will occur randomly throughout the SRAM device with uniform distribution.
- The PANSAT SRAM wash will instantly remove all errors in memory once per orbit and never during transit through the SAA.

Based on these assumptions, the number of SEUs expected after one SAA orbit is:

$$(27) \quad \frac{10^{-5} \text{ SEUs/bit/day}}{4 \text{ SAA orbits/day}} = 2.5 \times 10^{-6} \text{ SEUs/bit/SAA orbit}$$

**1. Using One EDAC Circuit to Provide a Six-bit Check Word for the 16-bit Data Words in the 512K SRAM.**

Given that errors occur randomly in the SRAM, the probability of no errors in any 22-bit word after one SAA orbit is:

$$(28) \quad p_0 = \binom{22}{0} (2.5 \times 10^{-6})^0 [1 - (2.5 \times 10^{-6})]^{22-0} = 0.999945001444$$

The probability of 1-bit error in any 22-bit data word after 1 SAA orbit is:

$$(29) \quad p_1 = \binom{22}{1} (2.5 \times 10^{-6})^1 [1 - (2.5 \times 10^{-6})]^{22-1} = 0.000054997113$$

Therefore, the probability of an uncorrectable error (2 or more bits in error) in a 22-bit word after one SAA orbit is:

$$(30) \quad 1 - (0.999945001444 + 0.000054997113) = 0.000000001443$$

The probability of uncorrectable errors in the SRAM space after one SAA orbit is:

$$(31) \quad 2^{18} \times 0.000000001443 = 0.000378273792$$

The expected number of SAA orbits between uncorrectable errors is:

$$(32) \quad \frac{1}{0.000378273792} \approx 2643$$

and the expected time between uncorrectable errors is:

$$(33) \quad \frac{2643 \text{ SAA orbits}}{4 \text{ SAA orbits/day}} \approx 660 \text{ days} = 1.8 \text{ years}$$

**2. Using Two EDAC Circuits to Provide a Five-bit Check Word for each Data Byte in the 512K SRAM.**

Given that errors occur randomly in the SRAM, the probability of no errors in any 13-bit word after one SAA orbit is:

$$(34) \quad p_0 = \binom{13}{0} (2.50 \times 10^{-6})^0 [1 - (2.50 \times 10^{-6})]^{13-0} = 0.999967500487$$

And the probability of 1-bit error in any 13-bit data word is:

$$(35) \quad p_1 = \binom{13}{1} (2.49 \times 10^{-6})^1 [1 - (2.49 \times 10^{-6})]^{13-1} = 0.0000324990250134$$

Therefore, the probability of an uncorrectable error (2 or more bits in error) in a 13-bit word after one SAA orbit is:

$$(36) \quad 1 - (0.999967500487 + 0.0000324990250134) = 0.000000000488$$

The probability of uncorrectable errors in the SRAM space after one SAA orbit is:

$$(37) \quad 2^{19} \times 0.000000000488 = 0.000255852544$$

The expected number of SAA orbits between uncorrectable errors is:

$$(38) \quad \frac{1}{0.000255852544} \approx 3908$$

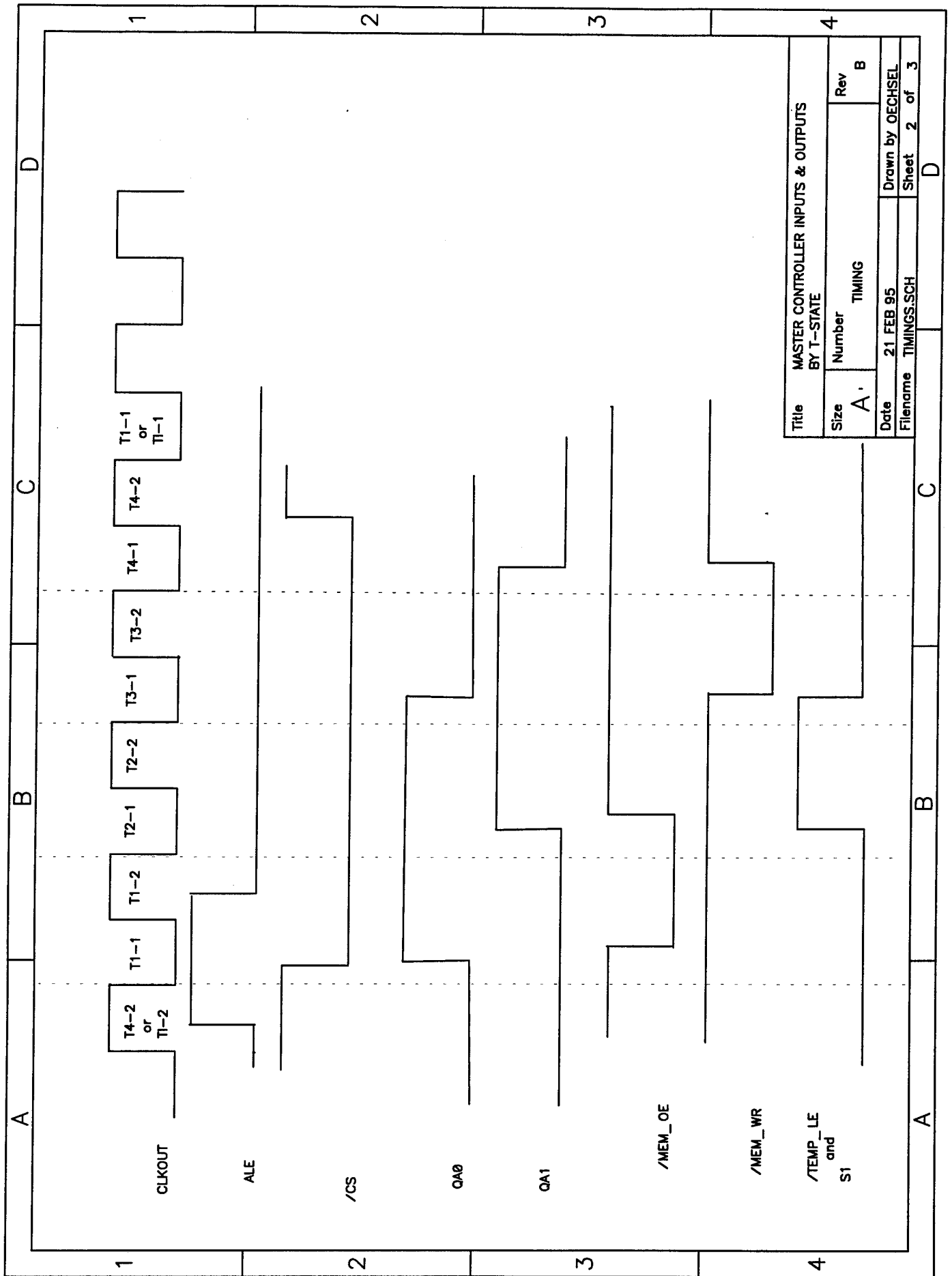
and the expected time between uncorrectable errors is:

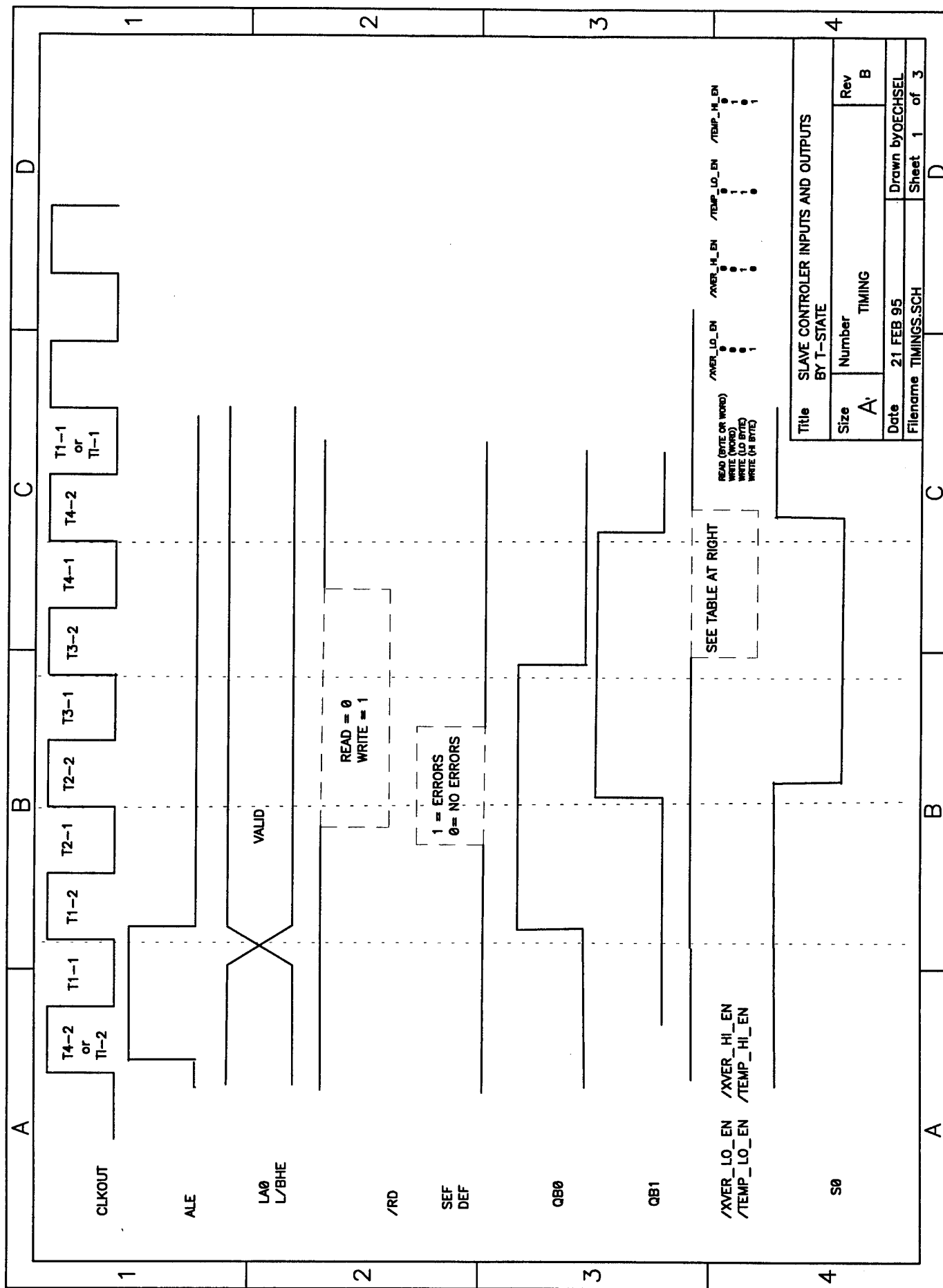
$$(39) \quad \frac{3908 \text{ SAA orbits}}{4 \text{ SAA orbits/day}} \approx 977 \text{ days} = 2.7 \text{ years}$$



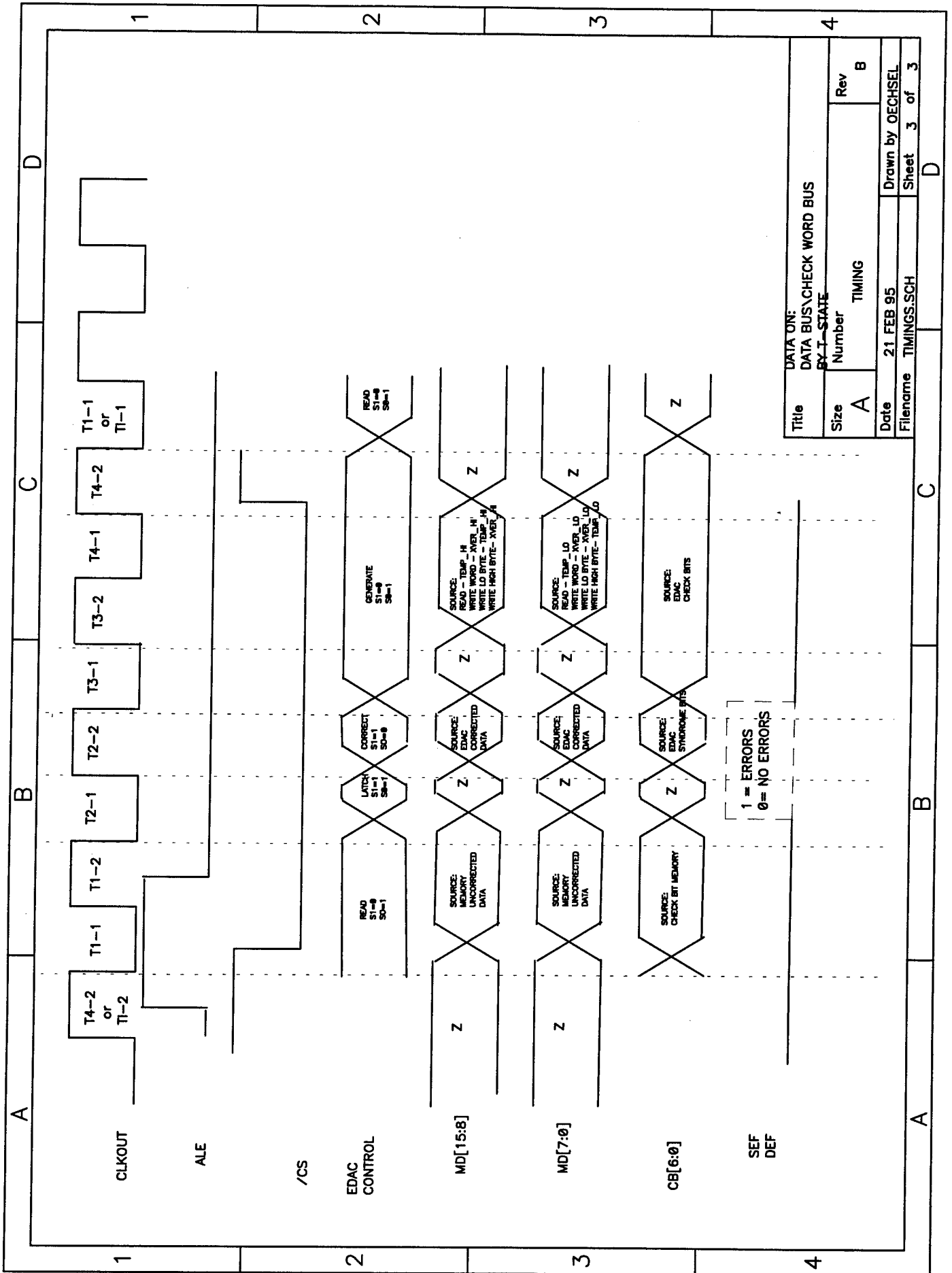
## **APPENDIX B. SYSTEM WAVEFORM DIAGRAMS**

- 1. Master State Machine Input and Output Waveforms**
- 2. Slave State Machine Output Waveforms**
- 3. Data Bus/Check Bit Bus Source by T-State Diagram**





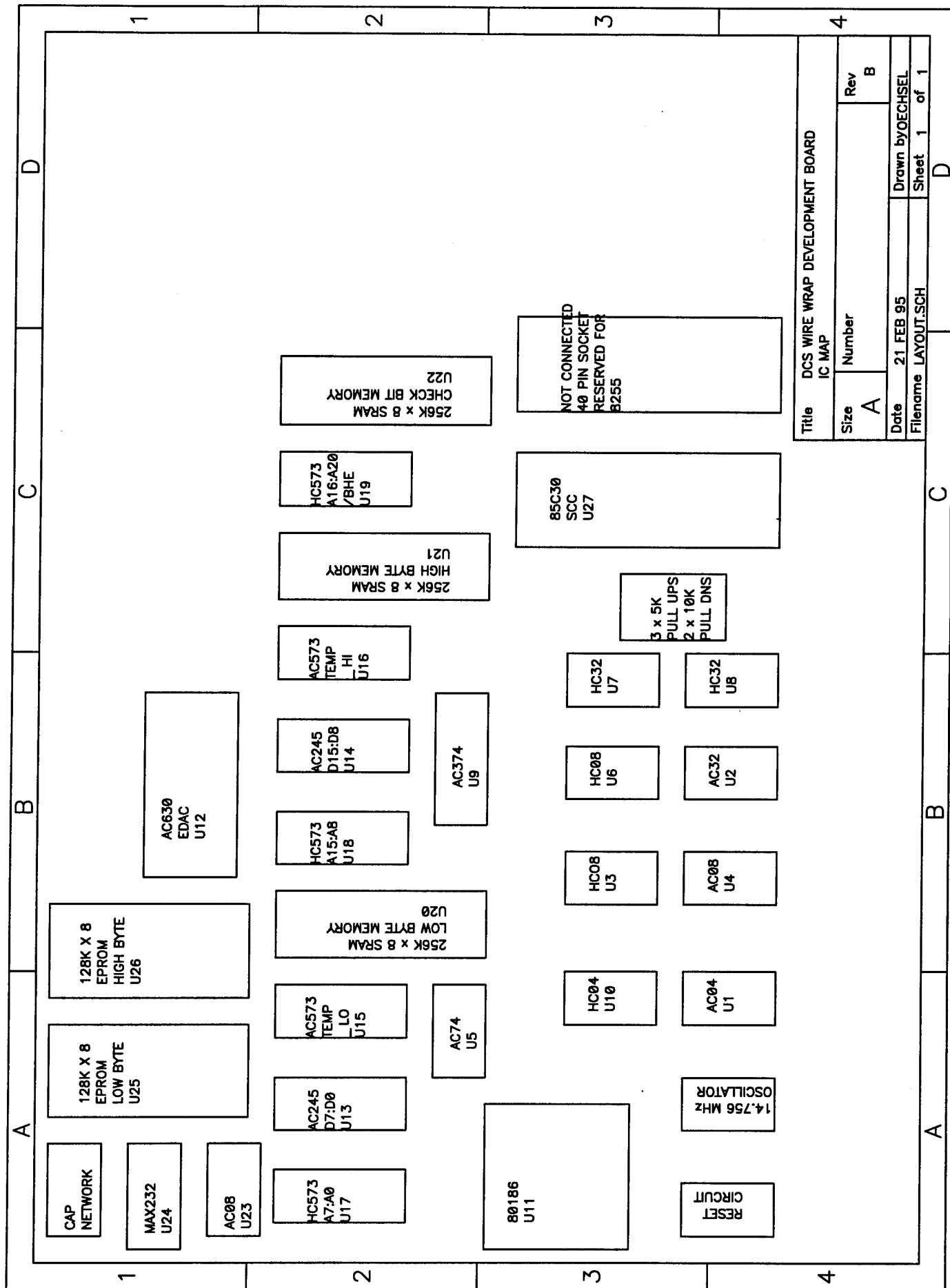
Title			
BY T-STATE			
Size	Number	Timing	
A		Rev B	
Date	21 FEB 95	Drawn by OECHSEL	
Filename	TIMINGS.SCH	Sheet	1 of 3



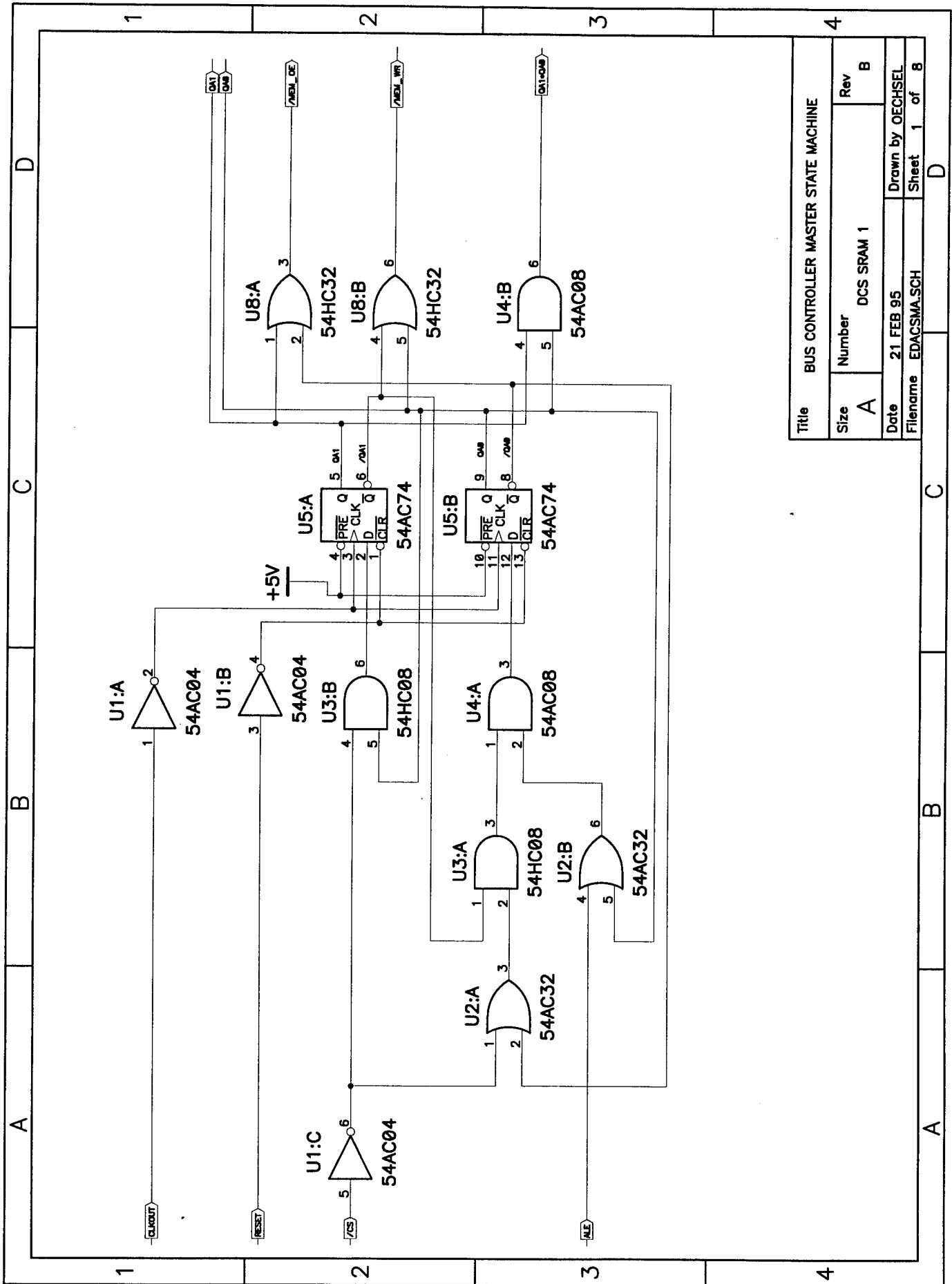


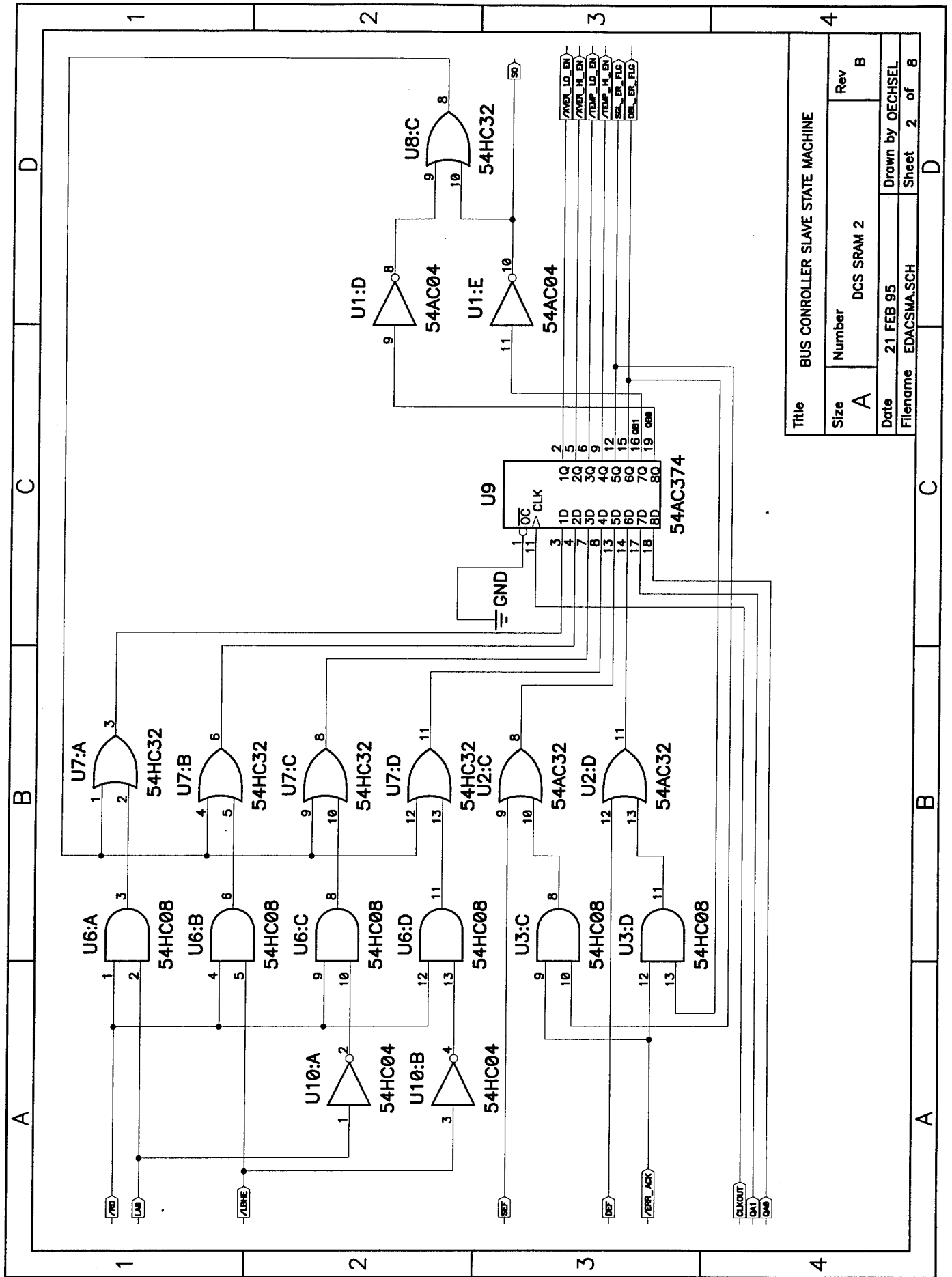
## **APPENDIX C. WIRE WRAP DEVELOPMENT BOARD SCHEMATICS**

- 1. IC Map**
- 2. M80C186XL and ACS630**
- 3. SRAM Bus Controller Master State Machine**
- 4. SRAM Bus Controller Slave State Machine and Error Flag Circuit**
- 5. Address Latches, SRAM Bus Transceiver and SRAM Temporary  
Registers**
- 6. Data and Check Word SRAM**
- 7. EEPROM**
- 8. Serial Communications Controller (SCC) and RS-232 Interface Circuit.**
- 9. Unused Devices**

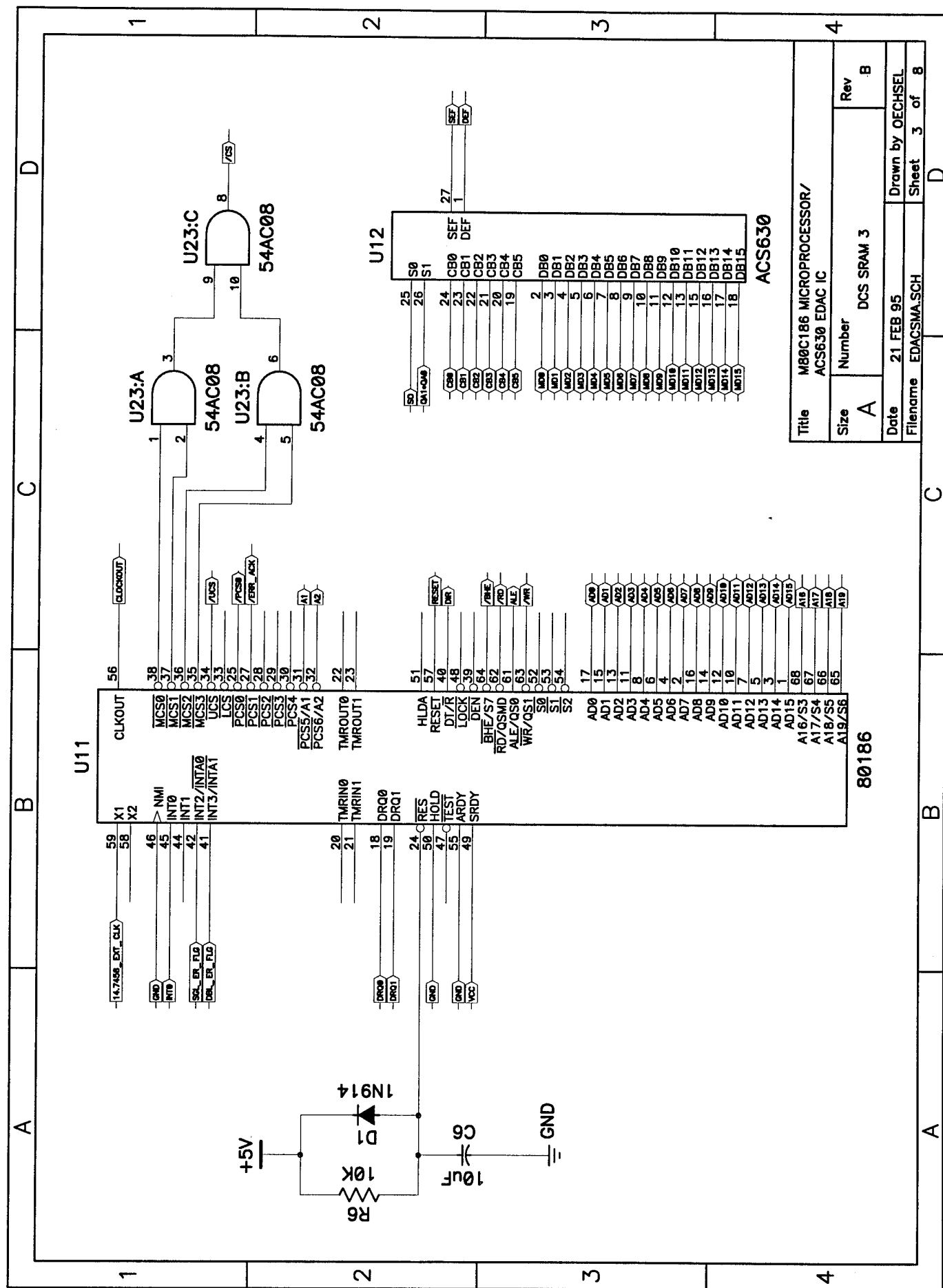


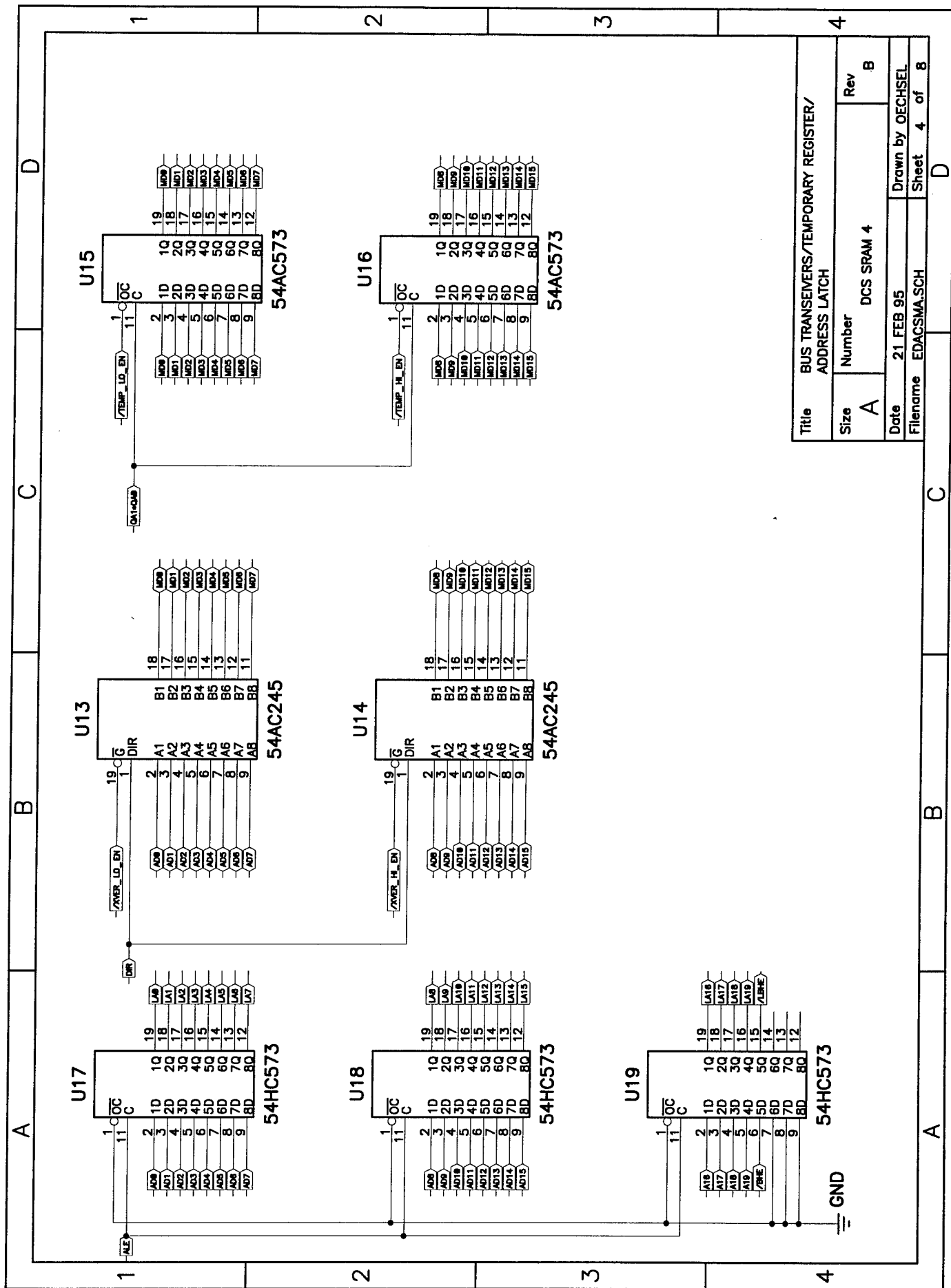
Title DCS WIRE WRAP DEVELOPMENT BOARD			
IC MAP			
Size	Number	Rev	
A		B	
Date	21 FEB 95		Drawn by QOECHSEL
Filename	LAYOUT.SCH	Sheet	1 of 1



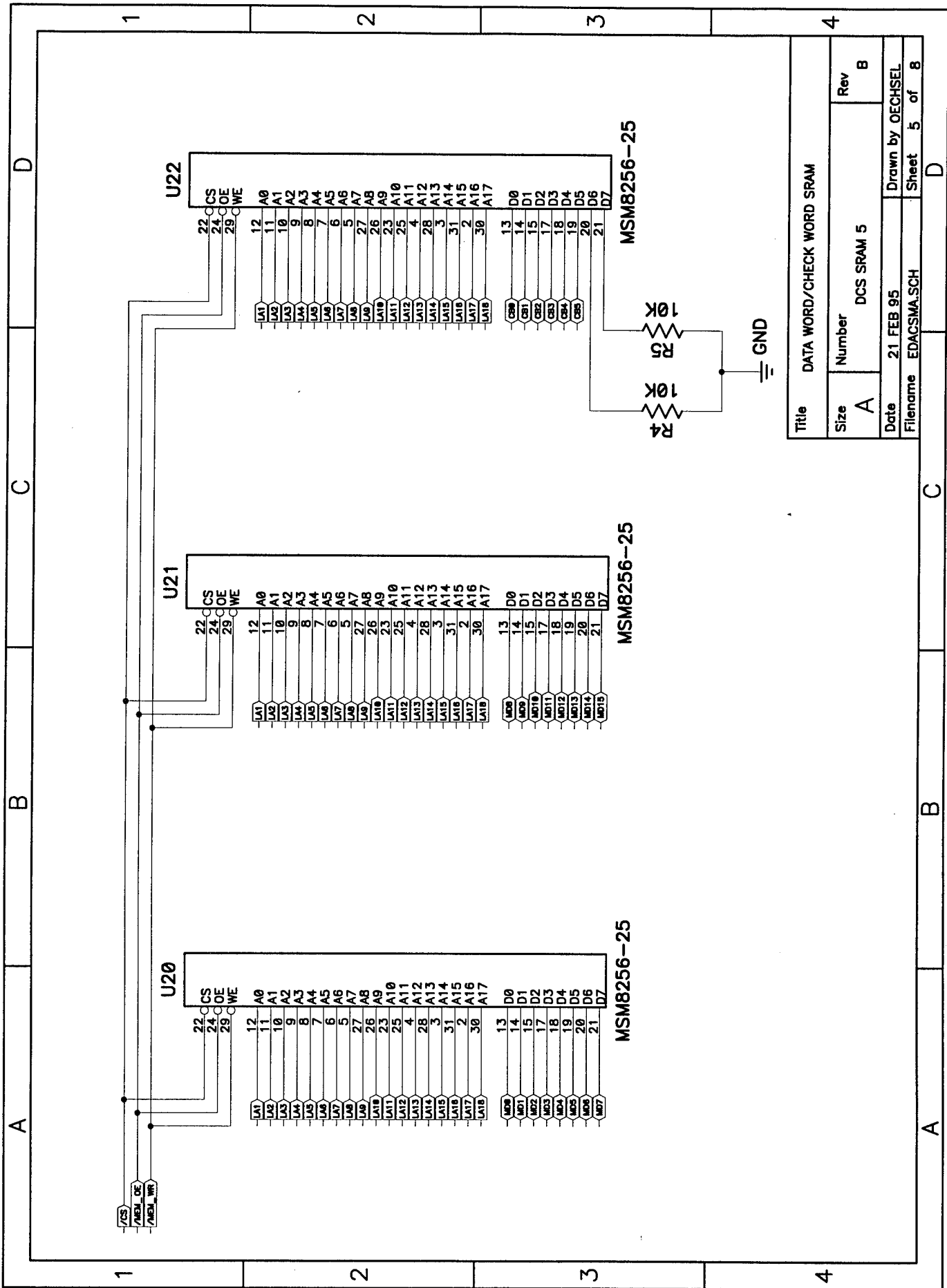


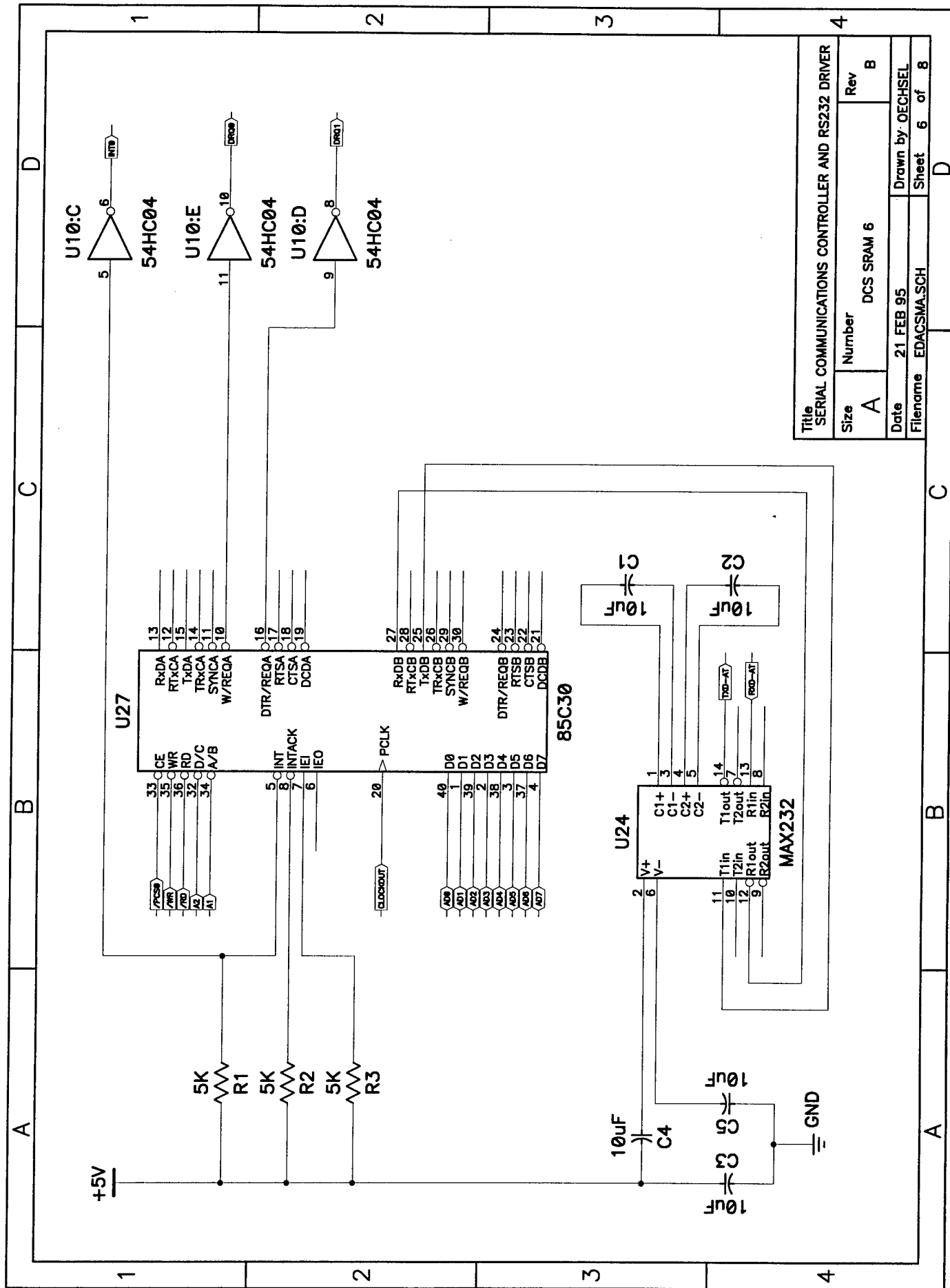
Title BUS CONTROLLER SLAVE STATE MACHINE			
Size A	Number DCS SRAM 2	Rev B	
Date 21 FEB 95	Drawn by OECHSEL	Sheet 2 of 8	
Filename EDACSMASCH			





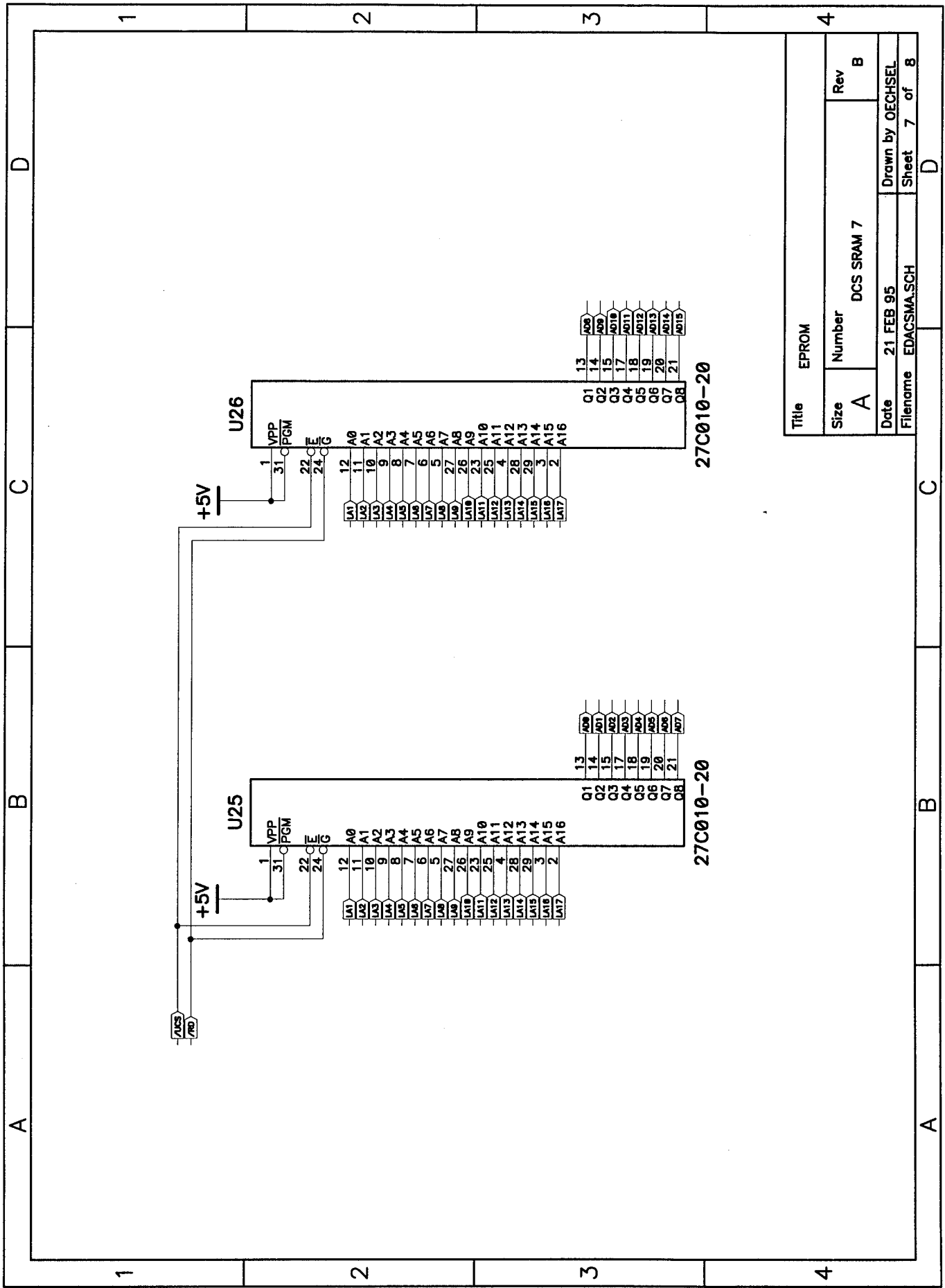
Title			
BUS TRANSCEIVERS/TEMPORARY REGISTER/ ADDRESS LATCH			
Size	Number	Rev	
A	DCS SRAM 4	B	
Date	21 FEB 95		
Filename		EDACSMASCH	
Date		Drawn by OECHSEL	
Sheet		4 of 8	





Title			
SERIAL COMMUNICATIONS CONTROLLER AND RS232 DRIVER			
Size	Number	Rev	
A	DCS SRAM 6	B	
Date		21 FEB 95	
Filename		EDACSMA.SCH	
		Sheet 6 of 8	





Title		EPROM	
Size	Number	DCS SRAM 7	Rev
A			B
Date	21 FEB 95		Drawn by OECHSEL
Filename	EDACSMASCH		Sheet 7 of 8



## **APPENDIX D. LIST OF NON-CRITICAL ADVANCED CMOS GATES**

The following is a list of Advanced CMOS (AC) logic gates, shown in the schematics in Figures 11 and 12 and Appendix C, that may be replaced with High-speed CMOS logic gates without affecting system operation based upon the manufacturers timing specifications for operation in the -40° to +85 °C range.

54AC04 Inverters:

U1:B

U1:D

U1:E

54AC32 2-input OR gate:

U2:A ( only if external SRAM chip-select signal logic requires less than 26.6 ns)



## LIST OF REFERENCES

1. Underwood, C. I. et al., "Space Science and Microsatellites - A Case Study: Observations of the Near-earth Radiation Environment Using the Cosmic-ray Effects and Dosimetry (CREDO) Payload On-board UoSAT-3," presented at the AIAA/USU Conference on Small Satellites, Logan, UT, August 1991.
2. "PANSAT Functional Requirements," SSD-S-SY-000, Naval Postgraduate School Space Systems Academic Group, February 21, 1995 Draft.
3. Johnson Space Center, NASA FAX, Subject: Space Shuttle Altitudes and Inclinations, April 1994.
4. Patterson, Sheila A., "Thermal Analysis of PANSAT Batteries and Electrical Power Subsystem," Master's Thesis, Naval Postgraduate School, Monterey, CA, 1994.
5. Messenger, G. C. and Ash, M. S., *The Effects of Radiation on Electronic Systems*, Van Nostrand Reinhold, New York, NY, 1991.
6. Koga, R., et al., "On the Suitability of Non-hardened High Density SRAMS for Space Applications," *IEEE Trans. on Nuclear Science*, Vol. 38, No. 6, pp. 1507-1511, 1991.
7. Adams, Leonard, et al., "Proton Induced Upsets in the Low Altitude Polar Orbit," *IEEE Trans. on Nuclear Science*, Vol. 36, No. 6, pp. 2339-2343, 1989.
8. *M80c186/C188 80C186XL/C188XL User's Manual*, Intel Corporation, Mt. Prospect, IL, 1992.
9. *Military and Special Products*, Vol. II, 1993, Intel Corporation, Mt. Prospect, IL, 1993.
10. *Product Selection Guide*, SG103, Harris Semiconductor Corporation, Melbourne, FL, 1994.
11. Leu, D. L., "Preliminary Digital Control System Design for the Petite Amateur Navy Satellite (PANSAT)," Master's Thesis, Naval Postgraduate School, Monterey, CA, 1994.
12. "54/74 AC/ACT Series: System Design," Harris AnswerFax File Number 7001, Harris Semiconductor Corporation, February 1995.
13. Harboe-Sørensen, R., et al., "The Behavior of Measured SEU at Low Altitude During Periods of High Solar Activity," *IEEE Trans. on Nuclear Science*, Vol. 37, No. 6, pp. 1938-1946, 1990.



## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center ..... 2  
     Cameron Station  
     Alexandria, Virginia 22304-6145
  
2. Library, Code 52 ..... 2  
     Naval Postgraduate School  
     Monterey, California 93943-5101
  
3. Department Chairman, Code EC ..... 1  
     Department of Electrical and Computer Engineering  
     Naval Postgraduate School  
     Monterey, California 93943-5121
  
4. CDR Randy L. Wight, Code SP/Wt ..... 1  
     Space Systems Academic Group  
     Naval Postgraduate School  
     Monterey, California 93943-5110
  
5. Dr. Herschel H. Loomis, Jr., Code EC/Lm ..... 1  
     Department of Electrical and Computer Engineering  
     Naval Postgraduate School  
     Monterey, California 93943-5121
  
6. Dr. Rudolph Panholzer, Code SP/Pz ..... 1  
     Chairman, Space Systems Academic Group  
     Naval Postgraduate School  
     Monterey, California 93943-5002
  
7. Dr. I. Michael Ross, Code AA/Ro ..... 8  
     Department of Aeronautical and Astronautical Engineering  
     Naval Postgraduate School  
     Monterey, California 93943-5106
  
8. Naval Research Laboratory ..... 1  
     Attn: Code 9110, LCDR Coutley  
     4555 Overlook Ave. SW  
     Washington, DC 20375
  
9. LT Craig R. Oechsel ..... 2  
     52 Highpoint Drive  
     Gulf Breeze, FL 32561